Advant® OCS
with Master software

Parameter Builder

Version 3.2

User’s Guide

3BSE013930R201

Use of **DANGER**, **WARNING**, **CAUTION**, and **NOTE**

This publication includes, **DANGER**, **WARNING**, **CAUTION**, and **NOTE** information where appropriate to point out safety related or other important information.

DANGER	Hazards which could result in severe personal injury or death
WARNING	Hazards which could result in personal injury
CAUTION	Hazards which could result in equipment or property damage
NOTE	Alerts user to pertinent facts and conditions

Although **DANGER** and **WARNING** hazards are related to personal injury, and **CAUTION** hazards are associated with equipment or property damage, it should be understood that operation of damaged equipment could, under certain operational conditions, result in degraded process performance leading to personal injury or death. Therefore, comply fully with all **DANGER**, **WARNING**, and **CAUTION** notices.

TRADEMARKS

Advant, AdvaCommand, AdvaInform, and AdvaBuild are registered trademarks of ABB Asea Brown Boveri Ltd., Switzerland.

NOTICE

The information in this document is subject to change without notice and should not be construed as a commitment by ABB Automation Products GmbH. ABB Automation Products GmbH assumes no responsibility for any errors that may appear in this document.

In no event shall ABB Automation Products GmbH be liable for direct, indirect, special, incidental or consequential damages of any nature or kind arising from the use of this document, nor shall ABB Automation Products GmbH be liable for incidental or consequential damages arising from use of any software or hardware described in this document.

This document and parts thereof must not be reproduced or copied without ABB Automation Products GmbH’s written permission, and the contents thereof must not be imparted to a third party nor be used for any unauthorized purpose.

The software described in this document is furnished under a license and may be used, copied, or disclosed only in accordance with the terms of such license.

Copyright © ABB Automation Products GmbH 1999

3BSE001264/E

Template: 3BSE001286/F

TABLE OF CONTENTS

Chapter 1 - Introduction

1.1 Before You Begin 1-1
1.2 Equipment Requirements 1-1
1.3 How to Use this Book 1-1
1.4 Conventions 1-2
1.5 Related Documentation 1-3
1.6 Release History 1-3
1.7 Terminology 1-3
1.8 Product Overview 1-3
    1.8.1 Product Version 1-3
    1.8.2 Product Structure 1-3
    1.8.3 Main Features 1-4
1.9 User Interface 1-6

Chapter 2 - Installation

2.1 General Information 2-1

Chapter 3 - Configuration/Application Building

3.1 Considerations 3-1
    3.1.1 Project and Directory Structures 3-1
3.2 Capacity & Performance 3-1
3.3 Database Configuration 3-2
3.4 Application Start-up 3-2
    3.4.1 How to Upgrade an Old Project 3-2
    3.4.2 How to Handle Parameter Aspects Within the Structure Builder 3-3
    3.4.3 Parameter Builder (Parameter Template) 3-6
        3.4.3.1 Navigation in the Template 3-7
        3.4.3.2 Search for Parameter Values 3-9
        3.4.3.3 Modify Attributes 3-12
        3.4.3.4 Synchronize Object Names and Reference Designations 3-14
        3.4.3.5 Close the Parameter Template 3-16
3.5 Application Tutorial 3-16
3.6 Application Procedures 3-17
    3.6.1 Creation of Project 3-17
    3.6.2 Configuration of Parameter Aspect Types 3-19
        3.6.2.1 Add New Aspect Type 3-21
        3.6.2.2 Delete Aspect Type 3-34
        3.6.2.3 Modify Existing Aspect Type 3-35

CONTENTS (continued)

3.6.2.4 Search for Aspect Type in Other Projects 3-37
3.6.2.5 Export Aspect Type Definition...................................... 3-40
3.6.2.6 Import Aspect Type Definition...................................... 3-41
3.6.2.7 Print Aspect Type Definition... 3-42
3.6.3 Local Parameter Builder Configuration 3-42
3.6.4 Configuration of Templates and Reports 3-45
3.6.5 User Defined Reports... 3-46
3.6.5.1 Reuse Project Global Data... 3-48
3.7 Configuration/Application Building Menus and Toolbars.................................... 3-51
3.7.1 Menus.. 3-51
3.7.1.1 Formsheet 3-51
3.7.1.2 Datasheet 3-53
3.7.1.3 Filter Dialog 3-54
3.7.1.4 Print Preview Dialog 3-55
3.7.1.5 Report Designer Dialog... 3-56
3.7.1.6 Forms Designer Dialog.. 3-56
3.7.2 Toolbars.. 3-56
3.7.2.1 Formsheet 3-56
3.7.2.2 Datasheet 3-57
3.7.2.3 Print Preview Dialog 3-58
3.7.2.4 Report Design Dialog... 3-58
3.7.2.5 Forms Design Dialog.. 3-58

Chapter 4 - Runtime Operation

4.1 Operating Overview 4-1
4.2 Runtime Tutorial 4-1
4.3 Operating Instructions.. 4-1
4.4 Runtime Operations Menus.. 4-1

Chapter 5 - Maintenance

5.1 Error Messages... 5-1
5.1.1 Database Error Messages 5-1
5.2 Reporting Problems... 5-6
5.3 What to Do at Program Halts... 5-6
5.4 Recommendations... 5-6

Appendix A - Format Definitions

A.1 Preconfigured Parameter Aspect Types A-1
A.1.1 Fieldbus Parameter Aspect Type.. A-1

CONTENTS (continued)

A.1.2 Station Parameter Aspect Type A-1
A.1.3 Board/ Module Parameter Aspect Type A-2
A.1.4 Signal Parameter Aspect Type A-2
A.1.5 Process Object Parameter Aspect Type A-3
A.2 How to Define References A-5
A.2.1 Relative Reference to the Same Object A-6
A.2.2 Relative Reference to Another (Related) Object A-7
A.2.3 Absolute Reference to Any Object A-9

Appendix B - Glossary of Terms

INDEX

ILLUSTRATIONS

Figure 1-1. Manual Organization 1-1
Figure 1-2. Example Motor Advant Object.................... 1-4
Figure 1-3. Example Signal Advant Object 1-5
Figure 1-4. Instantiation from a Library.................... 1-5
Figure 3-1. Object Window 3-3
Figure 3-2. New Aspect Dialog.................... 3-4
Figure 3-3. Select Parameter Type 3-4
Figure 3-4. Import Process: Instantiation from Library 3-6
Figure 3-5. Board Parameter Template: Formsheet View 3-7
Figure 3-6. Board Parameter Template: Datasheet View 3-8
Figure 3-7. Filter Buttons 3-8
Figure 3-8. Station Template: Design View 3-9
Figure 3-9. Filter Dialog.................... 3-10
Figure 3-10. Datasheet View with Filtered Board Parameter Aspects.................... 3-11
Figure 3-11. Sort Buttons 3-12
Figure 3-12. Find Dialog.................... 3-12
Figure 3-13. Replace Dialog 3-13
Figure 3-14. Replace NULL-Values.................... 3-13
Figure 3-15. Synchronize All Parameter Instances.................... 3-14
Figure 3-16. config.ini for Synchronize 3-15
Figure 3-17. Create Project - Configuration Selection Box 3-17
Figure 3-18. Create Project - Project Selection Box 3-18

ILLUSTRATIONS (continued)

Figure 3-19. Start Aspect Type Configuration 3-19
Figure 3-20. Aspect Type Configuration 3-20
Figure 3-21. Add New Aspect Type 3-21
Figure 3-22. Example: Aspect Instance of "AspectTypeA" 3-21
Figure 3-23. Select the Attribute Data Type 3-22
Figure 3-24. Parameter Aspect with Default Values 3-23
Figure 3-25. Example for User Defined Pick-lists 3-24
Figure 3-26. Select Source Type List 3-25
Figure 3-27. Select Source Type Table 3-26
Figure 3-28. Attach Pick-list Dialog (Initial State) 3-26
Figure 3-29. Dialog to Link Pick-list Database on Project Database 3-27
Figure 3-30. Link Tables Dialog 3-27
Figure 3-31. Dialog to Attach a Table on an Aspect Attribute 3-28
Figure 3-32. Aspect Type Attribute and Its Attached Table 3-28
Figure 3-33. Select Source Type Dependent on 3-29
Figure 3-34. Dialog to Select Dependent on Attributes 3-30
Figure 3-35. Aspect Type Attribute and Its Attached Table and Column 3-31
Figure 3-36. NewAspectType Parameter Template: Formsheet View 3-31
Figure 3-37. Add new Aspect Type Buttons 3-32
Figure 3-38. Copy Aspect Type 3-33
Figure 3-39. Delete Aspect Type 3-34
Figure 3-40. Modify Aspect Type 3-35
Figure 3-41. Keep Form and Reports Check Box 3-37
Figure 3-42. Search for Aspect Type 3-38
Figure 3-43. Details of the Selected Aspect Type 3-39
Figure 3-44. Compare Aspect Types 3-40
Figure 3-45. Export Aspect Type Definitions 3-40
Figure 3-46. Import Aspect Type Definitions 3-41
Figure 3-47. Print Definition 3-42
Figure 3-48. Local Parameter Builder Configuration 3-42
Figure 3-49. Toolbar Print Preview 3-45
Figure 3-50. Toolbar for Formsheet View of Aspect Type Templates 3-45
Figure 3-51. Create Delete Report Dialog 3-46
Figure 3-52. Preview Report Button 3-47
Figure 3-53. User Defined Report 3-48
Figure 3-54. Reuse Project Global Data Fields in Parameter Datasheet 3-49
Figure 3-55. Project Global Data Form 3-49
Figure 3-56. Project Global Data Save as Dialog 3-50

ILLUSTRATIONS (continued)

Figure 3-57. Formsheet Toolbar 3-56
Figure 3-58. Datasheet Toolbar 3-57
Figure 3-59. Print Preview Toolbar 3-58
Figure A-1. Example - Relative Reference to the Same Object........A-7
Figure A-2. Example - Parameter Reference to Another (Related) Child Object A-9
Figure A-3. Example - Absolute Parameter Reference to Any Object. A-10

TABLES

Table 1-1. Related Documents........ 1-3
Table 1-2. Release History........ 1-3
Table 5-1. Error Messages of Database Application 5-1
Table A-1. Fieldbus Parameter A-1
Table A-2. Station Parameter........A-1
Table A-3. Board/Module Parameter........A-2
Table A-4. Signal Parameter........A-2
Table A-5. Process Object Parameter........A-3

Chapter 1 Introduction

1.1 Before You Begin

This manual describes the Parameter Builder. The manual is meant to be used for to get familiar with the functions, commands and utilities of the Parameter Builder. To get the most recent hints, recommendations and settings please read the actual Release Notes which are delivered along with the product.

The reader of this manual should have proper education about engineering in general and the work flow usually used in Advant Engineering projects.

1.2 Equipment Requirements

To get the actual requirements and recommendations please read the actual Product Guide and Release Notes.

1.3 How to Use this Book

The following figure illustrates the organization of this user manual

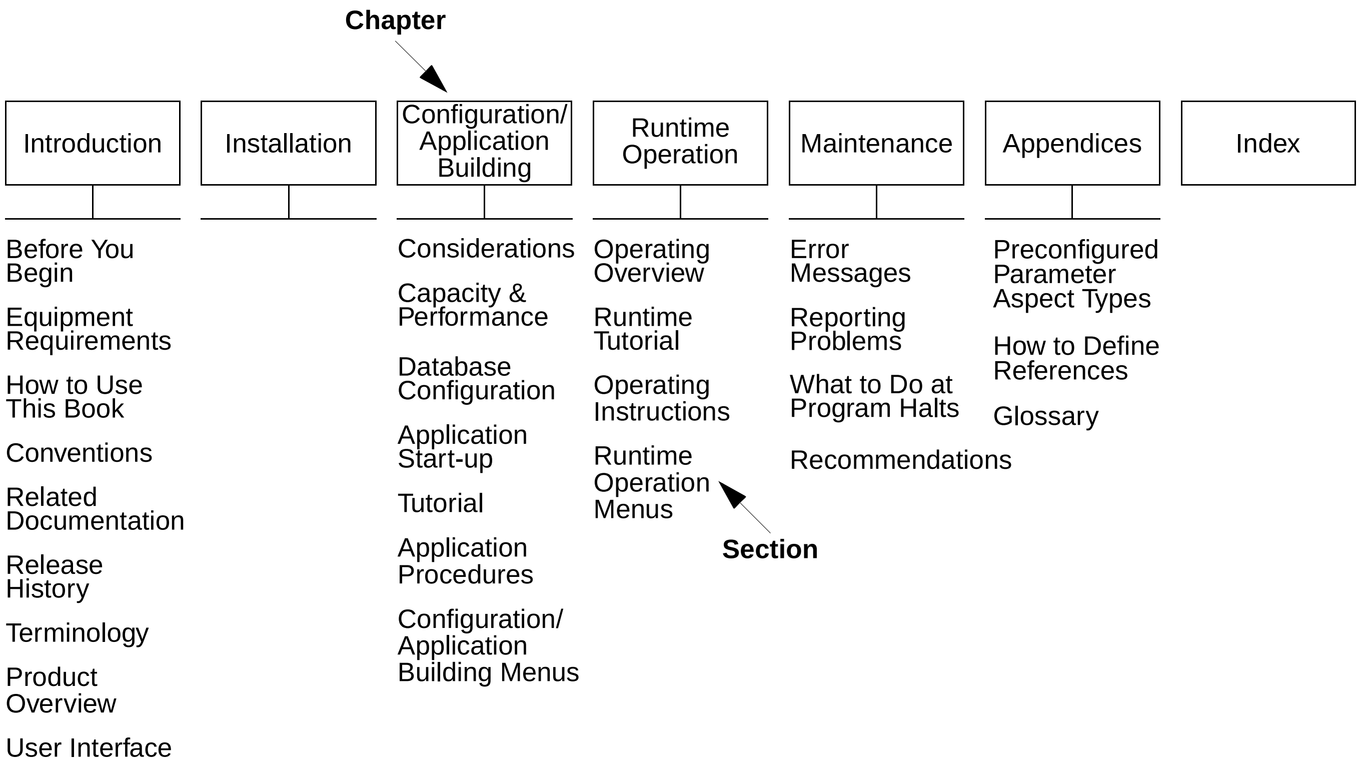


Figure 1-1. Manual Organization

1.4 Conventions

The following conventions are used throughout this manual for the presentation of material:

- Words in proper names of screen elements (for example, the title in the title bar of a window, the label for a field of a dialog box) are initially capitalized.
- Names of keyboard keys are boldfaced.
  - Initial capital letters are used for the name of a keyboard key if it is labeled on the keyboard. For example, the **Return** key or **<Return>**.
  - Lowercase letters are used for the name of a keyboard key that is not labeled on the keyboard. For example, the **space bar**, **comma key**, and so on.
  - In descriptions, the word "key" is used in conjunction with its name. For example, pressing the **Return** key initiates the process.
  - In tables and procedures, angle brackets replace the word "key." For example, press **<Return>**.
  - A dash is used to indicate that you must simultaneously press several keys. For example, **<Ctrl C>** indicates that you must hold down the **Control** key while pressing the **C** key.
  - The phrase "press and release" is used to indicate that you sequentially press several keys.
- Names of push and toggle buttons are boldfaced.
  - Initial capital letters are used for the name of a push or toggle button labeled on the display. For example, the **OK** button.
  - In descriptions, tables and procedures, the word "button" is used in conjunction with its name. For example, pressing the **OK** button initiates the process.
- Names of menus and menu items are boldfaced.
  - Initial capital letters are used for the name of menus listed in the menu bar. For example, the **File** menu.
  - Initial capital letters are used for the name of menu items. For example, to open a new window, choose **New** from the **File** menu.
- System prompts/messages are shown in the `Courier font`, and user responses/input are in the **`boldfaced Courier font`**. For example, if you enter a value out of range, the following message is displayed:

  ```
  Entered value is not valid. The value must be 0-30.
  ```

  In a tutorial, you may be told to enter the string TIC132 in a field. The string is shown as follows in the procedure:

  `TIC132`

1.5 Related Documentation

A list of Documentation related to Parameter Builder is shown in Table 1-1.

Table 1-1. Related Documents

Title	Description
Structure Builder User's Guide	This book describes the creation of projects, Advant objects, structures and aspects and how to navigate in projects
Engineering Methods Reference Manual	This book describes the workflow and the principles of object oriented engineering by using AdvaBuild engineering software
Bulk Data Handling and Reuse	This book describes the working with Structure Builder's Import/ Export Functions

1.6 Release History

This is the third product release of the Parameter Builder founded on the Advant object concept.

Table 1-2. Release History

Version	Description
3BSE013930R0001	Represents the Parameter Builder 3.0/0
3BSE013930R0101	Represents the Parameter Builder 3.1/0
3BSE013930R201	Represents the Parameter Builder 3.2/0

1.7 Terminology

See Appendix B, Glossary of Terms.

1.8 Product Overview

1.8.1 Product Version

This manual is related to Parameter Builder Release 3.2.

1.8.2 Product Structure

The Parameter Builder is composed of several SW packages:

- Parameter Builder including an interface to the Structure Builder
- Import package to insert customer bulk data prepared by the Instrument Index Data Acquisition Tool
- MS Access and predefined forms, reports and queries.

1.8.3 Main Features

Within the engineering process the Parameter Builder is used as data container of specific automation/ engineering data which are supposed to be used by other Builders around the Advant Engineering Workplace. The Parameter Builder provides user interfaces to include and to manipulate the engineering data.

So for example you might create an Advant object representing a motor comprising several aspects like (see Figure 1-2)

- A circuit containing a type circuit
- An Advant object identifier (Customer Id)
- A document
- A parameter aspect
- and so on.

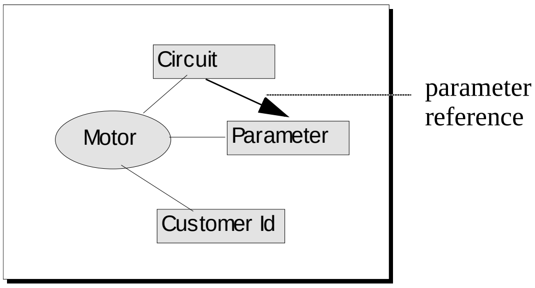


Figure 1-2. Example Motor Advant Object

Currently for type circuits (used as a container for either a control logic or a DBelement) and for documents (Word and Excel) there is a possibility to refer to data of Parameter aspects or Structure Builder aspects. A Parameter Reference is specified at the type circuit and may point to one or several parameter aspects of the same or other objects. The actual value the parameter reference points to is fetched into the type circuit on the user’s request. This is referred to as “updating the parameter reference”.

Another example is a signal object that comprises (see Figure 1-3)

- DBelement
- An Advant object identity (CustomerID)
- A parameter aspect.

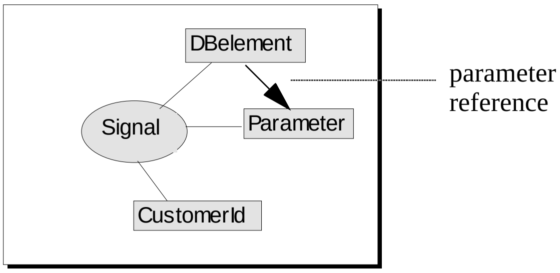


Figure 1-3. Example Signal Advant Object

In many cases the parameter values referred to by type circuits are provided by the customer. So the system allows you to import the customer data and store them as parameter aspects in the Parameter Builder. During the import of the customer data, objects predefined in the library are instantiated into the actual project - provided there is a reference into the library included in the data to be imported. By that all aspects predefined in the library like circuits, DBelements, structural information, parameter data, etc. will be taken over into the actual project. This advanced import functionality will improve your engineering efficiency by factors and result in a high quality of your engineering solution as you base on predefined typicals in a library.

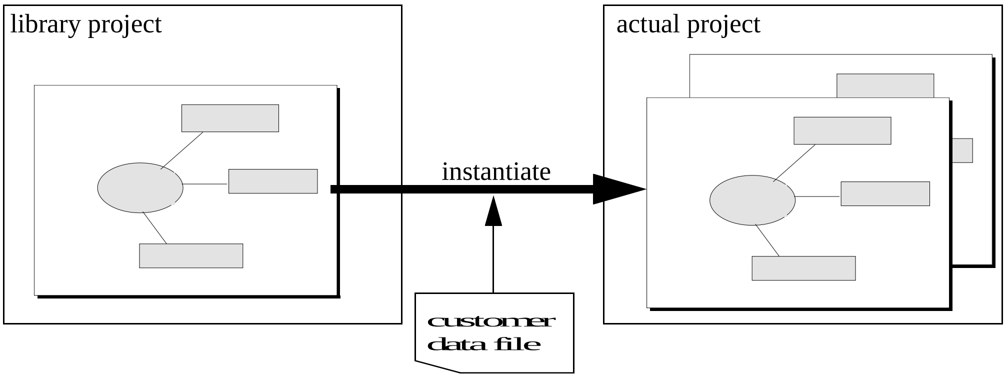


Figure 1-4. Instantiation from a Library

Moreover the values provided by the customer will overload the data predefined in the library during the import, thus including the required automation parameters into your system.
Of course after the import any value might be manipulated by the engineer.

As there are usually several versions of the customer data to be imported during the engineering process it is possible to re-import the customer data. Re-importing customer data will not damage your manual engineering work - except the values you'd like to import from the customer data will overwrite possible manual modifications you might have done before.

Once you have imported or manually created the parameter aspects the Parameter Builder enables you to efficiently handle the parameter values. Among other the following features are supported:

- Create and delete parameter aspects manually or by importing data from a file
- Navigate over parameter aspects
- Manipulate individual values or replace several values by one command
- Filter out certain parameter aspects fulfilling user definable filter criterias
- Printing parameter data by user definable reports
- etc.

In addition, the Parameter Builder contains the possibility to create Parameter Aspect Types with user definable Type names, attributes and data types.

For more information about:

- How to create and administrate the Parameter Aspect Types, please refer to Section 3.6.2, Configuration of Parameter Aspect Types.
- How to import data, please refer to "Balk Data Handling and Reuse Reference Manual".
- How to work with parameter aspects, please refer to the "Engineering Methods Reference Manual".
- How to specify parameter references in type circuits, please refer to the User's Guide "AMPL Control Configuration Commands for Structure Builder".

1.9 User Interface

The user interface of the Parameter Builder is implemented based on MS Windows. You choose the commands by accelerator keys and shortcuts on the keyboard or by mouse actions. The dialogs and forms which will appear on screen normally have Help buttons; clicking this button will open either the Parameter Builder's or the MS Office on-line help.

General information about how to use the MS Windows user interface is described for example in *Microsoft Windows NT 4.0* Help system.

Chapter 2 Installation

2.1 General Information

See Parameter Builder Release Notes.

Chapter 3 Configuration/Application Building

3.1 Considerations

3.1.1 Project and Directory Structures

Advant Engineering Server Part

There are four major areas on the server where the Parameter Builder stores data respectively applications:

- *abtool1*\par\release-id
  installed programs and configuration and help files of the Parameter Builder
- *advabld1*\data\project name \par\\
  storage of project specific data
- *advabld1*\data\project name\par\\db
  storage of the project specific database of the Parameter Builder
- *abtool1*\par\release\db
  pattern data base of the Parameter Builder used to create new projects
- *abtool1*\stb\release-id\mapping
  storage of the import mapping files

Legend:

abtool1	share name identifier of product path
advabld1	share name identifier of project path.

Advant Engineering Client Part

On the client there are configuration data stored like INI files etc. These files are stored (usually) on:

- **C:\\advabldenv**
- **C:\\tmp** will be used to store the trace file par.log if the trace mode is turned on.

Moreover there is one DLL placed onto the windows directory (usually) on:

- **C:\\Winnt\\system32**.

3.2 Capacity & Performance

The capacity of the system in terms of parameter aspects that can be stored is limited by the available disk space.

3.3 Database Configuration

Not applicable

3.4 Application Start-up

The Parameter Builder is an optional part of the AdvaBuild Engineering Software. It is accessed via the Structure Builder by opening an aspect which represents a parameter. There is one view provided in order to open a parameter aspect called “Attributes”. This view opens the Parameter Template presenting the various attributes of the selected parameter aspect.

Opening a parameter aspect is performed in the Object Window of the Structure Builder (see Figure 3-1).
How to start-up the Structure Builder and how to open its Object Window is described in the *Structure Builder User's Guide*.

The next sections describe:

- How to upgrade an older project (see Chapter 3.4.1, Configuration/Application Building)
- How to handle parameter aspects within the Structure Builder (see Chapter 3.4.2, Configuration/Application Building)
- How to handle parameter aspects within the Parameter Builder (Parameter Template - see Chapter 3.4.3, Configuration/Application Building).

3.4.1 How to Upgrade an Old Project

When you want to work with a project created with the previous release of Parameter Builder, this project has to be upgraded first.

Parameter Builder automatically recognizes an old project and suggests an upgrade before performing the following actions:

- Restore a project
- Add new aspect
- Delete aspect
- Open or print aspect
- Copy object containing parameter aspects
- Update aspect description.

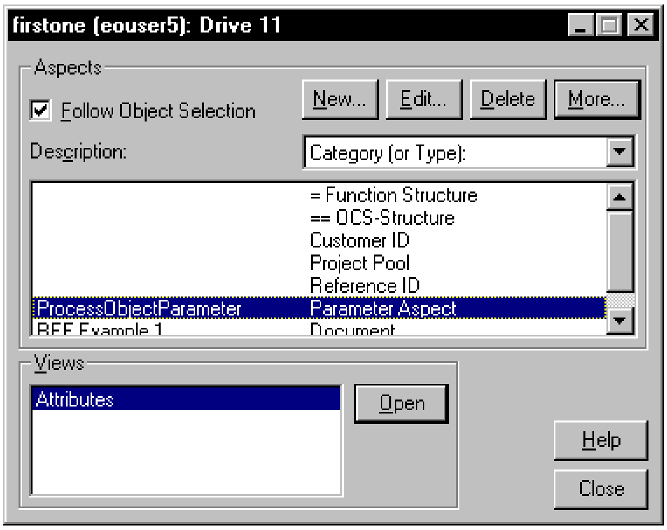
If the user chooses to upgrade the project, this is done automatically by **Parameter Builder**. On refusal, the requested action cannot be performed.

3.4.2 How to Handle Parameter Aspects Within the Structure Builder

Once opened the Object Window of the Structure Builder (see Figure 3-1) the user can perform several actions upon aspects. The following considerations are focusing on parameter aspects. The functions that can be performed on parameter aspects within the Structure Builder are:

- Add a new parameter aspect
- Delete a parameter aspect
- Print a parameter aspect
- Copy a parameter aspect
- Import parameter aspects
- Open a parameter aspect and by that start the Parameter Builder.

These functions will be described in the following subsections.



firstone (eouser5): Drive 11

☑ Follow Object Selection

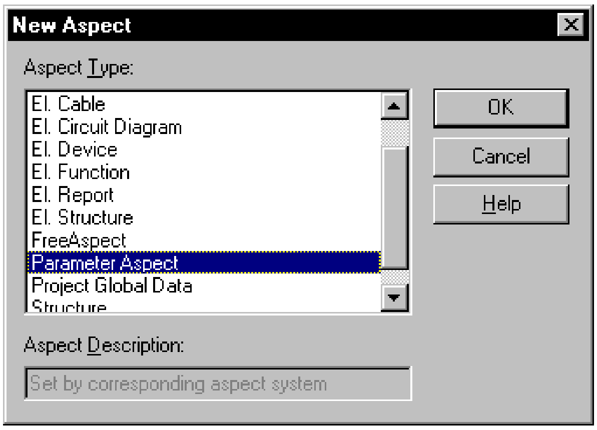
Description:	Category (or Type):
	= Function Structure
	== OCS-Structure
	Customer ID
	Project Pool
	Reference ID
ProcessObjectParameter	Parameter Aspect
BEF Example 1	Document

Views: Attributes

Figure 3-1. Object Window

Add a New Parameter Aspect

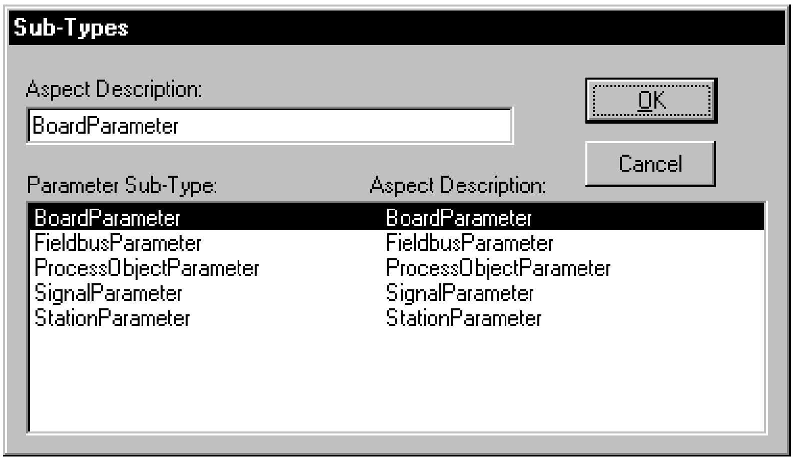
1. Press the **New** button in the Object Window,
   this will popup the “New Aspect dialog” (see Figure 3-2).
2. Select the parameter aspect type (there is only one kind of Parameter aspect,
   see Figure 3-3).
3. Press the **OK** button and the parameter aspect type selection window will appear.
4. Choose the desired type; it is recommended to enter an appropriated aspect description text.
5. Press the **OK** button and the parameter aspect will be inserted.



Aspect Type: Parameter Aspect

Aspect Description: Set by corresponding aspect system

Figure 3-2. New Aspect Dialog



Aspect Description: BoardParameter

Parameter Sub-Type:	Aspect Description:
BoardParameter	BoardParameter
FieldbusParameter	FieldbusParameter
ProcessObjectParameter	ProcessObjectParameter
SignalParameter	SignalParameter
StationParameter	StationParameter

Figure 3-3. Select Parameter Type

Delete a Parameter Aspect

Select the parameter aspect in the Object Window of the Structure Builder and press the **Delete** button.

Copy a Parameter Aspect

To copy an aspect please follows the steps:

1. Select in the Aspect Area the Parameter aspect to be copied.
2. Perform the menu entry **Copy** from the **Edit** menu.

3. Select in any structure window the Advant object who should get the aspect .
4. Perform the menu entry **Past** from the **Edit** menu.

To copy an object please follow the steps:

1. Select the object to be copied in the Structure Builder (e.g. in the Functional Structure).
2. Perform the menu entries **Copy** and **Paste** from the **Edit** menu.

Print a Parameter Aspect

1. Select the parameter aspect in the Object Window of the Structure Builder and press the **More** button or press the right button of your mouse.
2. Select the entry **Print** in the pop-up menu.
3. Select the view "Print" in the following pop-up menu.

This will start the Parameter Builder performing the print command.

Open a Parameter Aspect

Select a parameter aspect and the view "Attributes" in the Object Window of the Structure Builder and press the **Open** button.

This will start the Parameter Builder and open the Parameter Template (see Chapter 3.4.3, Configuration/Application Building) of the selected aspect. The Parameter Template displayed depends on the selection in the Structure Builder.

NOTE

Open and Print actions can also be started from the aspect area in the Structure Builder.

Import Parameter Aspects

Parameter aspects can be imported from files. The data to be imported is typically edited in Excel and saved in a comma delimited file format. The Parameter Aspect Types currently supported are described in Section A.1, Preconfigured Parameter Aspect Types. To import a file a dialog in the Structure Builder must be filled in. During the import of the file contents among others parameter aspects are created in the Parameter Builder.

As a prerequisite for the instantiation of parameter aspects, there must be an Advant object defined in the library which is referred to during the import. This library object serves as a template for the instantiation. All parameter aspects (and other aspects) defined for this object, will be instantiated into your actual project.

NOTE

Make sure, that the Parameter aspect types used in the library Advant objects are also created in your actual project.

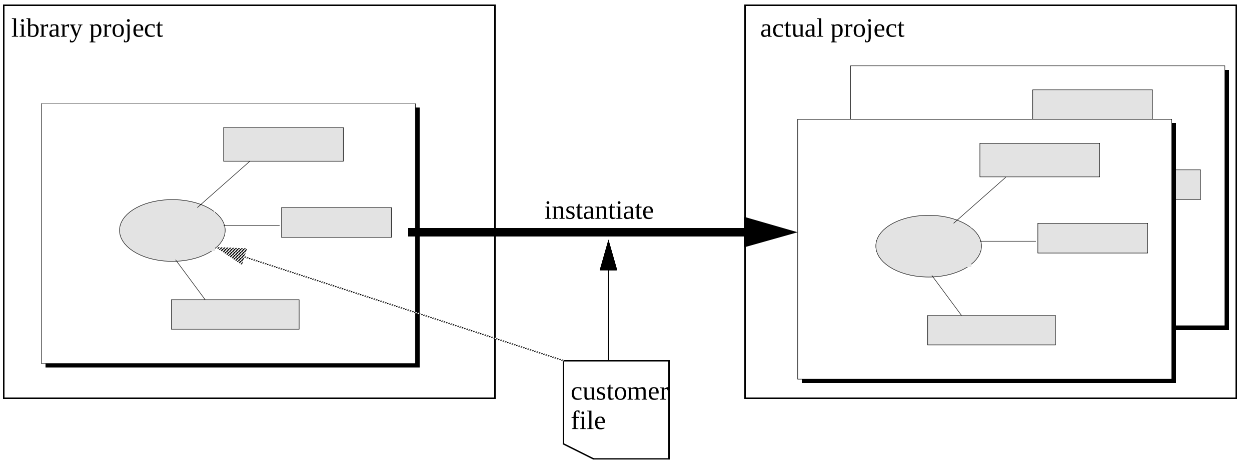


Figure 3-4. Import Process: Instantiation from Library

There can be several Advant objects predefined in the library incl. a parameter aspect. Depending on the library reference included in the data to be imported the related parameter aspects are instantiated.
The parameter aspects created will also be filled by the data provided in the import file.
If parameter aspects already exist at the time of the import they will be updated by the data provided in the import file.

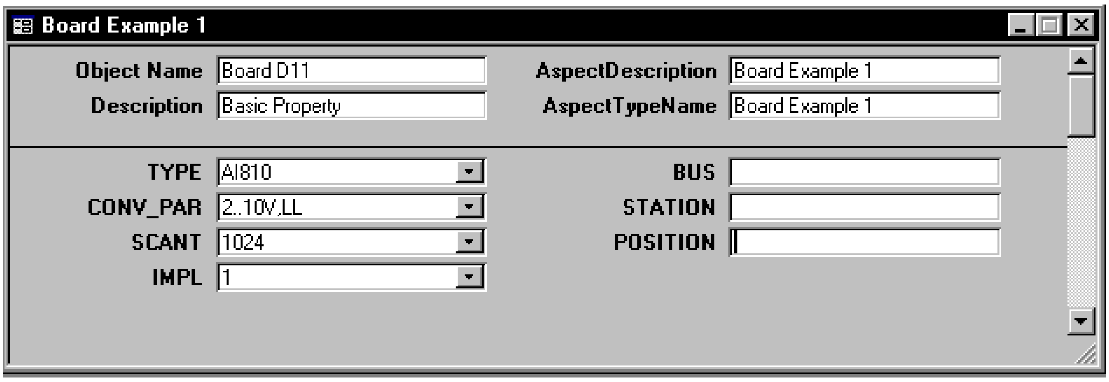
For more information:

- Please refer to Section 1.8, Product Overview and to the “Bulk Data Import and Reuse Reference Manual”.
- About the user interface for the file import, please refer to the *Structure Builder User’s Guide*.
- About the import of data as seen from the process (work flow) point of view, please refer to the *Engineering Methods User’s Guide*.

3.4.3 Parameter Builder (Parameter Template)

What’s the Parameter Template all about?

A parameter aspect is composed of several attributes. The Parameter Template is an MS Access based user interface (MS Access Form) to depict and modify the values of the attributes. Depending on the type of the parameter aspect there are different templates. They are automatically created during Aspect Type creation.



Object Name	Board D11	AspectDescription	Board Example 1
Description	Basic Property	AspectTypeName	Board Example 1
TYPE	AI810	BUS	
CONV_PAR	2..10V,LL	STATION	
SCANT	1024	POSITION	
IMPL	1		

Figure 3-5. Board Parameter Template: Formsheet View

The information presented is grouped into two sections:

1. Header part
   general information about the parameter aspect e.g. the Advant object name, the aspect description, the aspect type name, etc.
2. Body part
   Parameter Attributes, the various attributes of the parameter aspect

3.4.3.1 Navigation in the Template

There are two presentation views for the data and one design view:

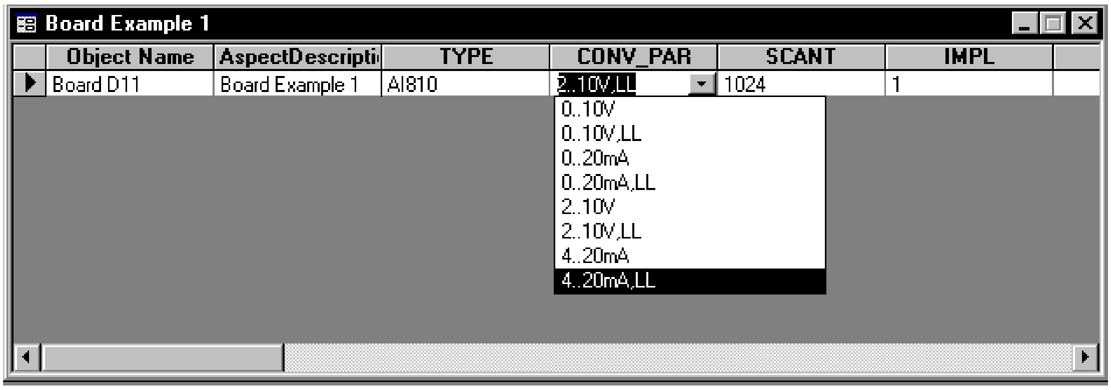
- Formsheet
- Datasheet
- Design.

Formsheet

When opening a Parameter Template in the Structure Builder the Formsheet (see Figure 3-5) is presented as the default view. Within the Formsheet you can navigate using the mouse or the **Tab** key

Datasheet

The datasheet is a tabular presentation of all attributes of a parameter aspect. Each line represents a parameter aspect.



Board Example 1

Object Name	AspectDescripti	TYPE	CONV_PAR	SCANT	IMPL
Board D11	Board Example 1	AI810	2..10V,LL	1024	1

0..10V
0..10V,LL
0..20mA
0..20mA,LL
2..10V
2..10V,LL
4..20mA
4..20mA,LL

Figure 3-6. Board Parameter Template: Datasheet View

The datasheet is opened by pressing the **Apply** F**ilter** button in the toolbar

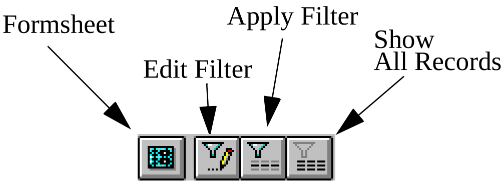


Figure 3-7. Filter Buttons

The parameter aspects presented in the datasheet depend on the current filter setting (see below). If you don’t define a filter the parameter aspect displayed is the one selected in the Structure Builder

To switch back to the formsheet press the **Formsheet** button in the toolbar

Within the datasheet you can scroll trough all attributes and all parameter aspects (if several parameter aspects are filtered) by using the horizontal and vertical scroll bars. You can also re-format the layout of the datasheet in terms of:

- hide and re-display attributes (columns)
- freeze and un-freeze one or several attributes
  This allows you to scroll horizontally through all attributes without moving the frozen attributes out of the window.
- set the column width and height.

These functions above can be selected from the menu “**Format**”.

Design

The design is used to customize the template. Modifications should only be made to the datafields (see the area which is marked and labeled with **1**); here you may rearrange, enlarge and rename the field names in a way you want. Do not manipulate the two other marked areas.

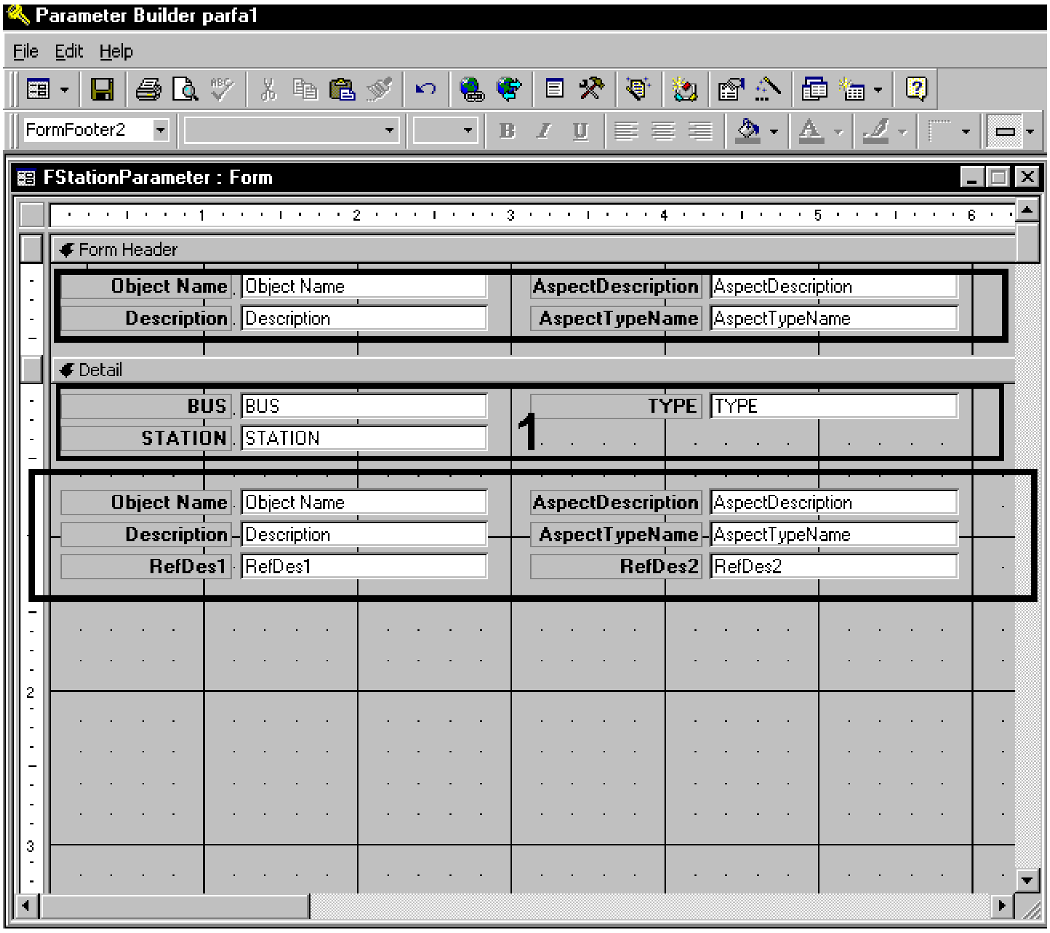


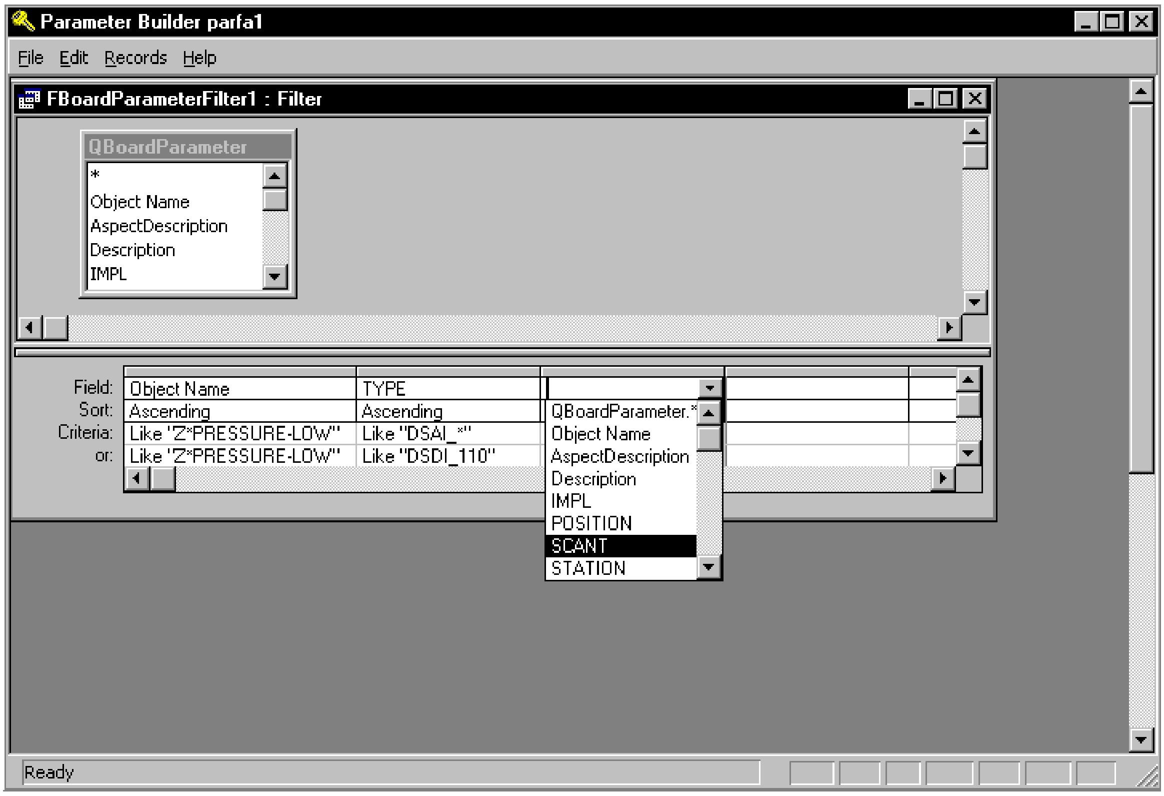
Figure 3-8. Station Template: Design View

3.4.3.2 Search for Parameter Values

Besides the possibilities provided in the Structure Builder the Parameter Builder allows you to search for parameter aspects according to user definable Filter criterias. The following steps have to be performed:

1. First open the Parameter Template using an arbitrary parameter aspect (see Chapter 3.4.2, Configuration/Application Building).
2. Press the **Edit Filter** button in the toolbar or select the menu **Edit Filter** from the **Records** menu.

This will launch the following filter dialog:



Field:	Object Name	TYPE	
Sort:	Ascending	Ascending	
Criteria:	Like "Z*PRESSURE-LOW"	Like "DSAI_*"	
or:	Like "Z*PRESSURE-LOW"	Like "DSDI_110"	

Figure 3-9. Filter Dialog

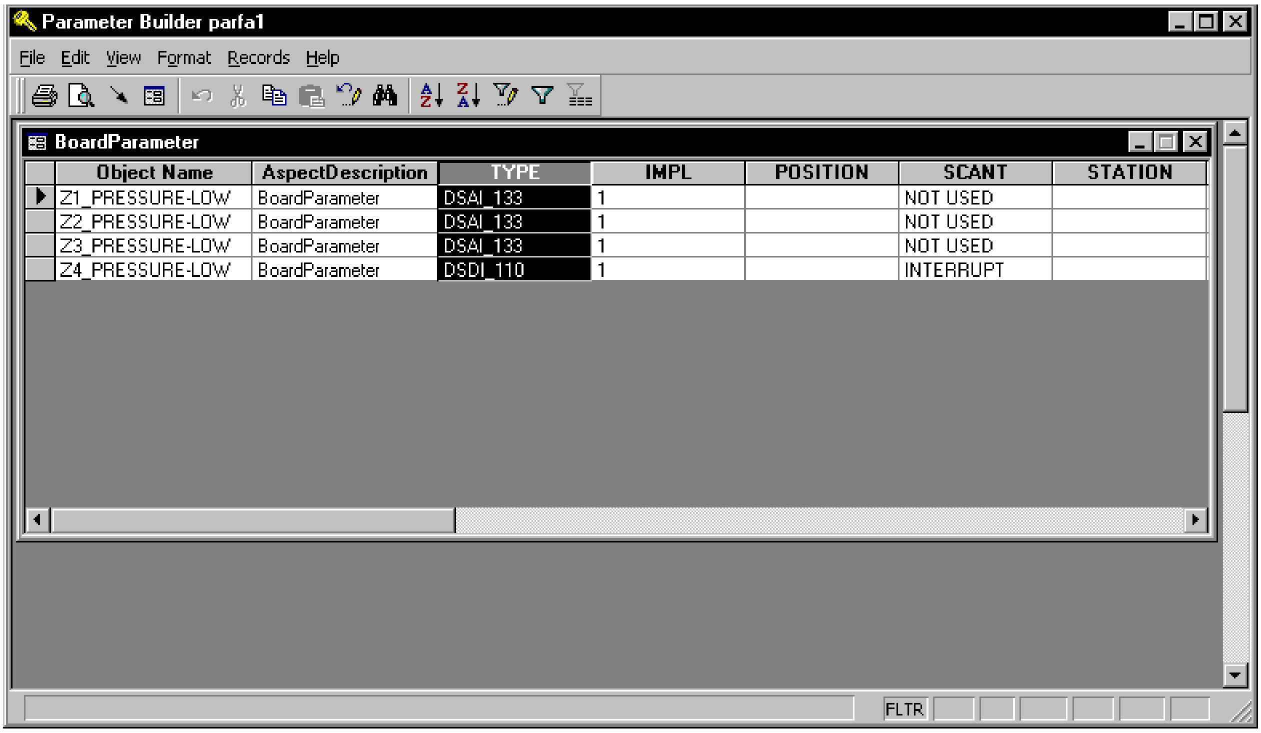
3. Select an attribute (field) from the selection box you would like to define a criteria for.

4. Enter the criteria for this field.
   You may also specify wildcards (*) in the criteria string,
   you may also enter several values which are put in a logical "OR" operation.
   For example see the entries in the first column of Figure 3-9:
   the objects to search for have to start with "Z" followed by any string (*) and terminated with "PRESSURE-LOW".

5. You can include all attributes in your filter specification. Each field specified will be added by a logical "AND" Operation to the whole filter specification.
   For example see the entries in the second column of Figure 3-9:
   the parameter aspects to search for must have the column "TYPE" set to "DSAI_*" or "DSDI_*"

   So finally all parameter aspects of all objects starting with "Z" and terminated with "PRESSURE-LOW" and having a board type of either "DSAI_*" or "DSDI_*" will be retrieved.

6. When you have completed the filter specification you can apply the filter by pressing the **Apply Filter** button in the toolbar. This results in displaying the datasheet view showing all parameter aspect fulfilling the filter specification.
   If you want to redefine the filter (if for example the hit rate is too high) just perform the same procedure again and adapt your filter criterias.
7. If you'd like to save the filter settings for later reuse select the entry **Save as Query** in the right hand mouse button context sensitive menu.
   A saved query can be reloaded next time by selecting the entry **Load from Query** from context sensitive menu.
8. If you want to see all parameter aspects press the **Show All Records** button in the toolbar or select the menu **Show All Records** from the **Records** menu



Object Name	AspectDescription	TYPE	IMPL	POSITION	SCANT	STATION
Z1_PRESSURE-LOW	BoardParameter	DSAI_133	1		NOT USED	
Z2_PRESSURE-LOW	BoardParameter	DSAI_133	1		NOT USED	
Z3_PRESSURE-LOW	BoardParameter	DSAI_133	1		NOT USED	
Z4_PRESSURE-LOW	BoardParameter	DSDI_110	1		INTERRUPT	

Figure 3-10. Datasheet View with Filtered Board Parameter Aspects

If there are many parameter aspects retrieved it might be sensible to sort them for the ease of navigation. In this case place the cursor in the field to be sorted and press the "**Sort Ascending**" or **Sort Descending** button from the toolbar or select the menu entry **Sort** from the **Records** menu.

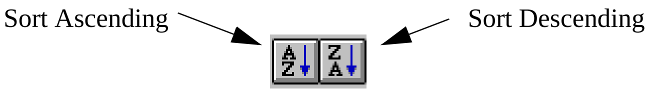
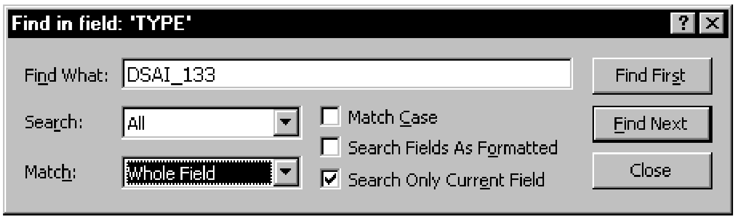


Figure 3-11. Sort Buttons

You can also specify more complex sort criterias as for example sort by "NAME" and within the "NAME" column sort by "BOARD_TYPE". This can be entered within the "Edit Filter" dialog by selecting "ascending" or "descending" within the row titled "sort". In the example depicted in Figure 3-9 the data retrieved is sorted first by the object name (ascending) and within the object name according to the board type (ascending). So the sort criteria is evaluated from left to right.

In addition to a filter specification you can search for certain parameter aspects within the current selection. This is done by specifying a Find Criteria in the following find dialog. The dialog is opened by selecting the menu **Find** from the **Edit** menu or by pressing the **Find** button in the toolbar


Find in field: 'TYPE'

Find What: DSAI_133

Search: All

Match: Whole Field

☐ Match Case

☐ Search Fields As Formatted

☑ Search Only Current Field

Figure 3-12. Find Dialog

Find What: text string to be found

Search: Up, Down or All can be chosen

Match: Whole Field, Any Part of Field or First Part of Field can be chosen

Match Case: distinguish capital and lower characters

Search Fields as formatted: search for formatted text (typically date formats). For more information please refer to the user manual of MS Access

Search Only Current Field: search in the current attribute (field) of all parameter aspects depicted in the datasheet or in all attributes. The current attribute is the one the cursor is located in.

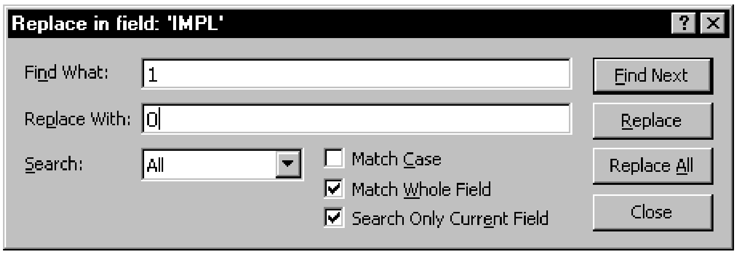
3.4.3.3 Modify Attributes

All attributes can be modified except:

- Object name
- Aspect description
- Aspect type.

The modification can be performed in both views (formsheet and datasheet).

If one or more attributes of several parameter aspects have to be modified there is a Replace Function as depicted in the following figure. The dialog is entered by selecting the menu entry **Replace** from the **Edit** menu



Replace in field: 'IMPL'

Find What: 1

Replace With: 0

Search: All

☐ Match Case

☑ Match Whole Field

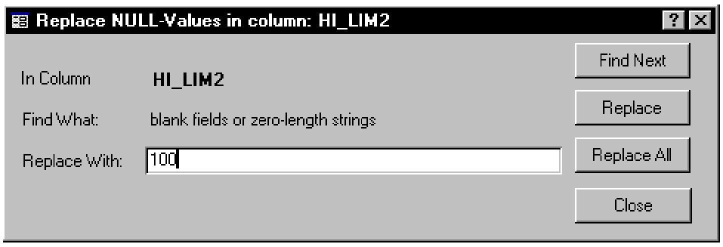
☑ Search Only Current Field

Figure 3-13. Replace Dialog

For example if you want to replace the value of field "IMPL" from "1" to "0" please follow the steps:

1. **Find What**: fill in the text to be replaced.
2. **Replace With: f**ill in the text to replace the old text.
3. **Search**: Up, Down or All can be chosen.
4. **Match Case**: distinguishes between upper and lower case characters.
5. **Match Whole Field**: if the text to be replaced is only a substring in the related field do not press this button.
6. **Search Only Current Field**: replace the old string only in the current field (attribute) the cursor is located in (in the example: Field IMPL) or in all fields (attributes).

To replace NULL values of a certain field use the Replace NULL Values dialog. The menu item is presented if the datasheet form is activated. To open dialog select menu entry **Replace NULL-Values** from the **Edit** menu.



Replace NULL-Values in column: HI_LIM2

In Column **HI_LIM2**

Find What: blank fields or zero-length strings

Replace With: 100

Figure 3-14. Replace NULL-Values

For example if you want to replace the NULL values of field “HI_LIM2” please follow the steps:

1. **Replace With: f**ill in the text to replace the NULL values.
2. Use **Find Next/ Replace** button to replace selectively.
3. Use **Replace All** to replace the values in one step.

3.4.3.4 Synchronize Object Names and Reference Designations

When you Open or Print a Parameter aspect instance only the

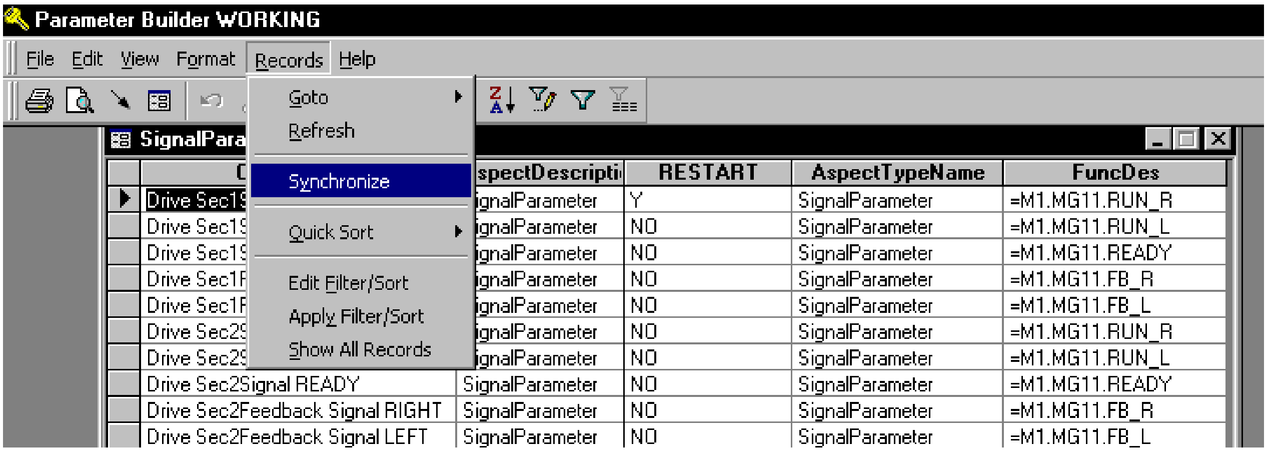
- object name and
- reference designations (Func, Loc, Mount, ContrDes)

of the object to which a Parameter instance belongs are updated from the **Structure Builder** database, whereas the aspect description of Parameter aspects is always up to date automatically.

As a result, it is possible that of all the Parameter aspect instances stored in **Parameter Builder** some have actual values for the object name and the reference designations and some have not which is particularly irritating when scanning all the Parameter instances in the datasheet view.

To synchronize the object names and reference designations for all Parameter instances:

Select menu item **Synchronize** from menu **Records** in Datasheet form.



C	spectDescripti	RESTART	AspectTypeName	FuncDes
Drive Sec1S	ignalParameter	Y	SignalParameter	=M1.MG11.RUN_R
Drive Sec1S	ignalParameter	NO	SignalParameter	=M1.MG11.RUN_L
Drive Sec1S	ignalParameter	NO	SignalParameter	=M1.MG11.READY
Drive Sec1F	ignalParameter	NO	SignalParameter	=M1.MG11.FB_R
Drive Sec1F	ignalParameter	NO	SignalParameter	=M1.MG11.FB_L
Drive Sec2S	ignalParameter	NO	SignalParameter	=M1.MG11.RUN_R
Drive Sec2S	ignalParameter	NO	SignalParameter	=M1.MG11.RUN_L
Drive Sec2Signal READY	SignalParameter	NO	SignalParameter	=M1.MG11.READY
Drive Sec2Feedback Signal RIGHT	SignalParameter	NO	SignalParameter	=M1.MG11.FB_R
Drive Sec2Feedback Signal LEFT	SignalParameter	NO	SignalParameter	=M1.MG11.FB_L

Figure 3-15. Synchronize All Parameter Instances

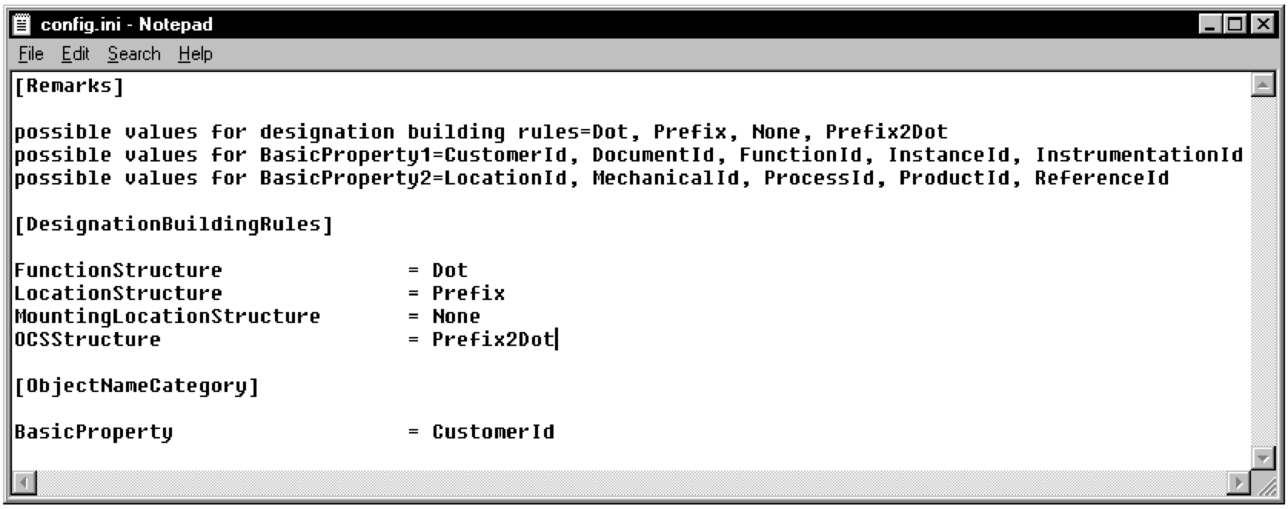
NOTE

Please note that execution of the Synchronize function might take quite some time, depending on the number of Parameter instances contained in the project.

Building Rules: Reference Designations

The format of the reference designations is determined by building rules. These building rules can be configured for every type of designation by editing a file *config.ini* in the project's database directory *advabld1*\data\<project>\par\db. Figure 3-16 shows an example of the **config.ini** file.

Before activating the **Synchronize** function for the first time, the *config.ini* file does not yet exist. You can either write it yourself using a normal editor or you can leave the creation of *config.ini* to the **Synchronize** function. In this case, *Dot* is used as a building rule for all four designations. Also, if entries in *config.ini* have no valid value due e.g. to a typing error, *Dot* is used as building rule.



```
[Remarks]

possible values for designation building rules=Dot, Prefix, None, Prefix2Dot
possible values for BasicProperty1=CustomerId, DocumentId, FunctionId, InstanceId, InstrumentationId
possible values for BasicProperty2=LocationId, MechanicalId, ProcessId, ProductId, ReferenceId

[DesignationBuildingRules]

FunctionStructure               = Dot
LocationStructure               = Prefix
MountingLocationStructure       = None
OCSStructure                    = Prefix2Dot

[ObjectNameCategory]

BasicProperty                   = CustomerId
```

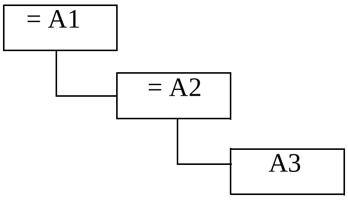
Figure 3-16. config.ini for ***Synchronize***

Possible values for the building rule are

- **Dot**
- **Prefix**
- **None**
- **Prefix2Dot**.

Example

If you have a hierarchy as follows:



the different building rules for the bottom object would result in the following:

Dot:	=A1.A2.A3
Prefix:	=A1=A2A3
None:	=A1A2A3
Prefix2Dot:	=A1.A2A3

Building Rule: Object Name

In the same initialization file the category or Basic Property for the object's name attribute is determined. Possible values for the category are:

Customer ID, Document ID, Function ID, Instance ID, Instrumentation ID, Location ID, Mechanical ID, Process ID, Product ID, and Reference ID.

The default for the Basic Property is **Customer ID**.

3.4.3.5 Close the Parameter Template

Close

If you select the menu **Close** from the **File** menu the template will be closed but the Parameter Builder application is still active. This allows a much faster start-up when opening the Parameter Template again.

Exit

If you select the menu **Exit** from the **File** menu the Parameter Builder application is totally terminated.

3.5 Application Tutorial

not applicable

3.6 Application Procedures

3.6.1 Creation of Project

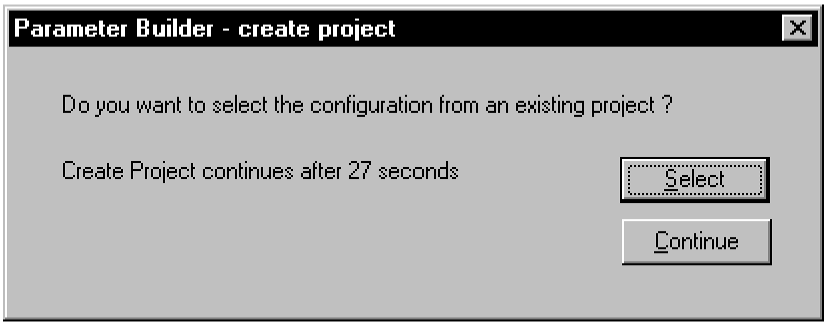
To create a new project start the administration utility through the Structure Builder, menu **Project** item **Administration** follow the dialogs and enter the required data (see *Structure Builder User's Guide*).

By default during creation of a project Parameter Builder copies a pattern project database into the new project location. This pattern database contains Parameter type configuration (Parameter type definitions, Templates, Reports etc.) for

- Board/Module Parameter
- Signal Parameter
- Station Parameter
- Process Object Parameter
- Fieldbus Parameter.

If you have already projects with your own Parameter type configuration you can take over this database configuration instead the default pattern database.

During the creation process, a dialog box is displayed (Figure 3-17) asking if the Parameter Builder database should be copied from an existing project. If this is not requested the **Continue** button can be pressed. This dialog box disappears after 30 seconds automatically and the project creation process continues with the standard behavior.

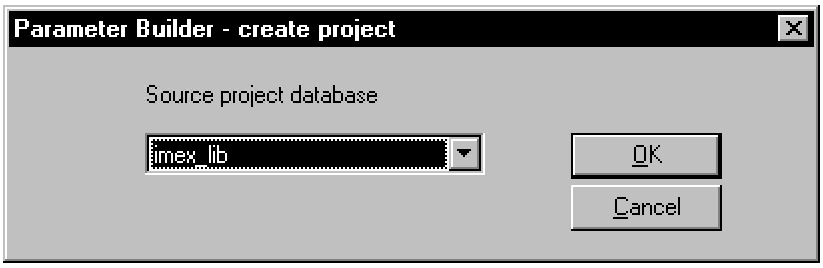

Parameter Builder - create project

Do you want to select the configuration from an existing project ?

Create Project continues after 27 seconds

Figure 3-17. Create Project - Configuration Selection Box

If it is requested to copy the database configuration from an existing project, press the **Select** button. At this a 2nd dialog box is displayed (Figure 3-18) where the source project can be selected from all existing projects. Select a project and press the **OK** button.



Source project database: imex_lib

Figure 3-18. Create Project - Project Selection Box

NOTE

Once you have started your project without the Parameter Builder and now you want to add it. To register the Parameter Builder to your project afterwards use the Structure Builder, menu **Project** item **Administration** (for details see *Structure Builder User’ Guide*).
If you copy an Advant object (which includes beside others also a Parameter Aspect) from a source project into your target project - where on target project the Parameter Builder was not registered - the aspects of the Advant object will be copied up to but excluding the Parameter Aspect without any warning.

3.6.2 Configuration of Parameter Aspect Types

The Parameter Builder offers a comfortable way to create up to 28 Parameter Aspect Types for a project. The dialog **Aspect Type Configuration** is useful to

- Add new Aspect Types manually or copy existing Aspect Types from other projects
- Modify existing Aspect Types and their Aspect Type attributes
- Delete unneeded Aspect Types
- Search for actual Aspect Type in other projects and compare those Aspect Types with ones from where the search was started
- Export of Aspect Type definition data to for example Instrument Index Data Acquisition tool as ASCII-file
- Import Aspect Types definition data from ASCII-File and create the Aspect Types based on the definition in the file
- Print Aspect Type definitions.

The AdvaBuild Parameter Builder menu bar, menu **Edit**, menu item **Aspect Type Configuration...** starts the corresponding dialog.

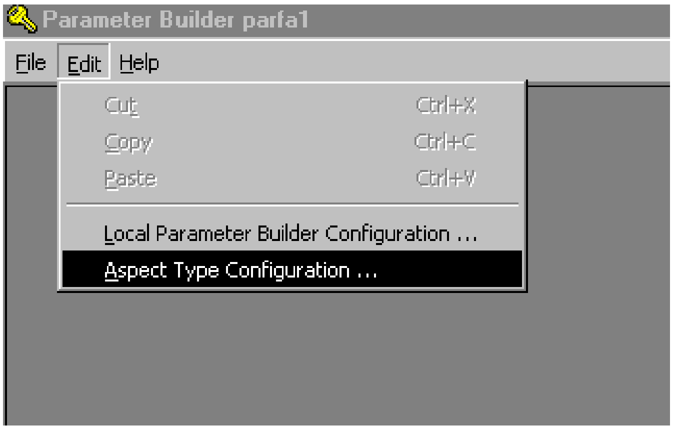

Parameter Builder parfa1
File
Edit
Help
Cut
Ctrl+X
Copy
Ctrl+C
Paste
Ctrl+V
Local Parameter Builder Configuration ...
Aspect Type Configuration ...

Figure 3-19. Start Aspect Type Configuration

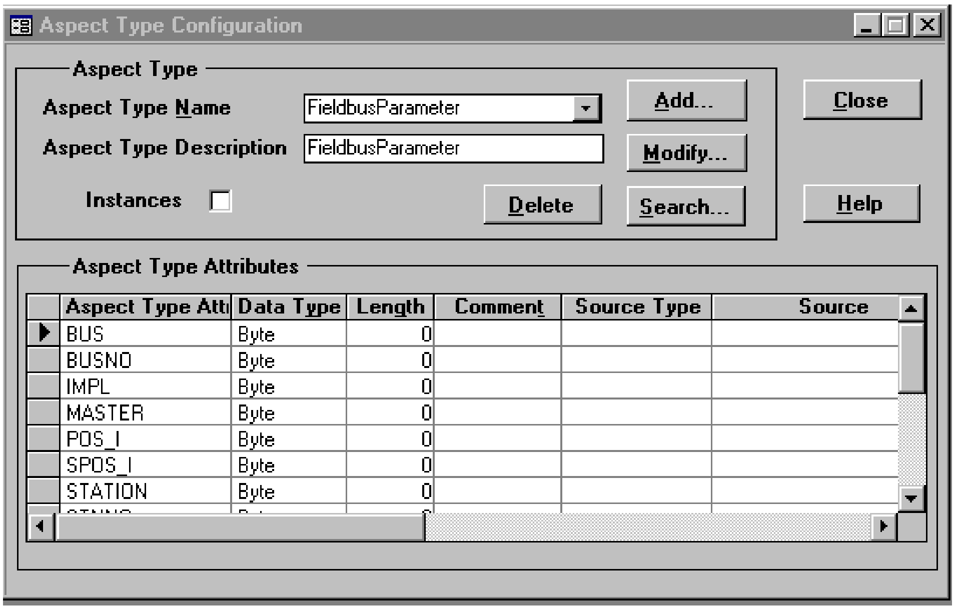
Select Aspect Types and Show Their Aspect Type Attributes

The window displays the Aspect Type of the previous configuration-session and it’s Aspect Type attributes, the Aspect Type name and description is displayed too.

The field **Instances** informs the user, that the specified Aspect Type has instances, if it is marked. In this case, the **Delete** button is invisible.

NOTE

It is recommended that Aspect Type Configuration activities is doing only one user on one project at the same time; it is not possible to import data if Aspect Type Configuration is working.



Aspect Type Name: FieldbusParameter

Aspect Type Description: FieldbusParameter

Instances ☐

Aspect Type Att	Data Type	Length	Comment	Source Type	Source
BUS	Byte	0			
BUSNO	Byte	0			
IMPL	Byte	0			
MASTER	Byte	0			
POS_I	Byte	0			
SPOS_I	Byte	0			
STATION	Byte	0			

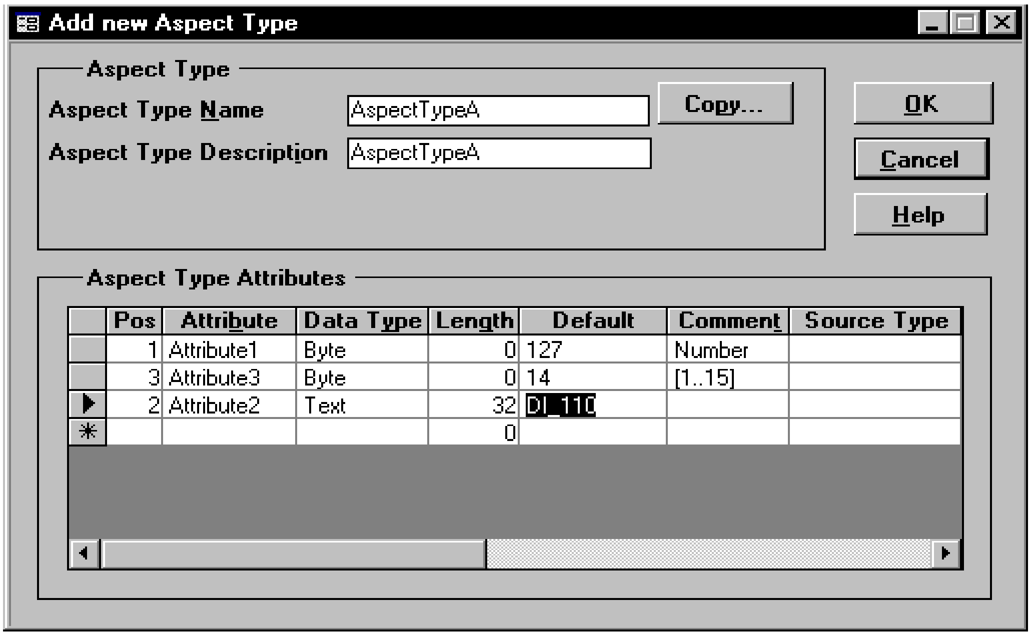
Figure 3-20. Aspect Type Configuration

The list-field **Aspect Type Name** shows a list of the defined Aspect Types within this project. The selection of a new Aspect Type name from the list displays the actual Aspect Type description and the Aspect Type attributes for this Aspect Type.

The shown buttons and their operations:

- **Close** closes the "Aspect Type Configuration" dialog.
- **Help** shows the help-window of the Parameter Builder .
- **Delete** allows to delete the current Aspect Type and related templates, reports. This button is not visible, if the actual Aspect Type has instances.
- **Add...** opens the "Add new Aspect Type" dialog (see Chapter 3.6.2.1).
- **Modify...** starts the "Modify existing Aspect Type" dialog (see Chapter 3.6.2.3).
- **Search...** starts "Search Aspect Type in other projects" dialog (see Chapter 3.6.2.4).

3.6.2.1 Add New Aspect Type



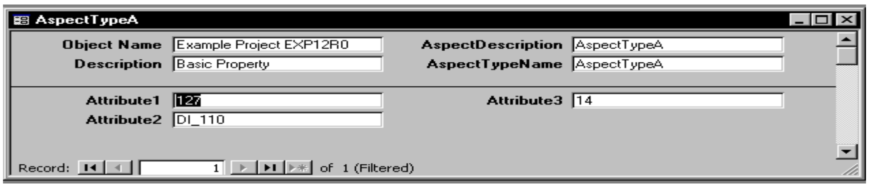
Aspect Type Name: AspectTypeA
Aspect Type Description: AspectTypeA

	Pos	Attribute	Data Type	Length	Default	Comment	Source Type
	1	Attribute1	Byte	0	127	Number	
	3	Attribute3	Byte	0	14	[1..15]	
▶	2	Attribute2	Text	32	DI_110		
✳				0			

Figure 3-21. Add New Aspect Type

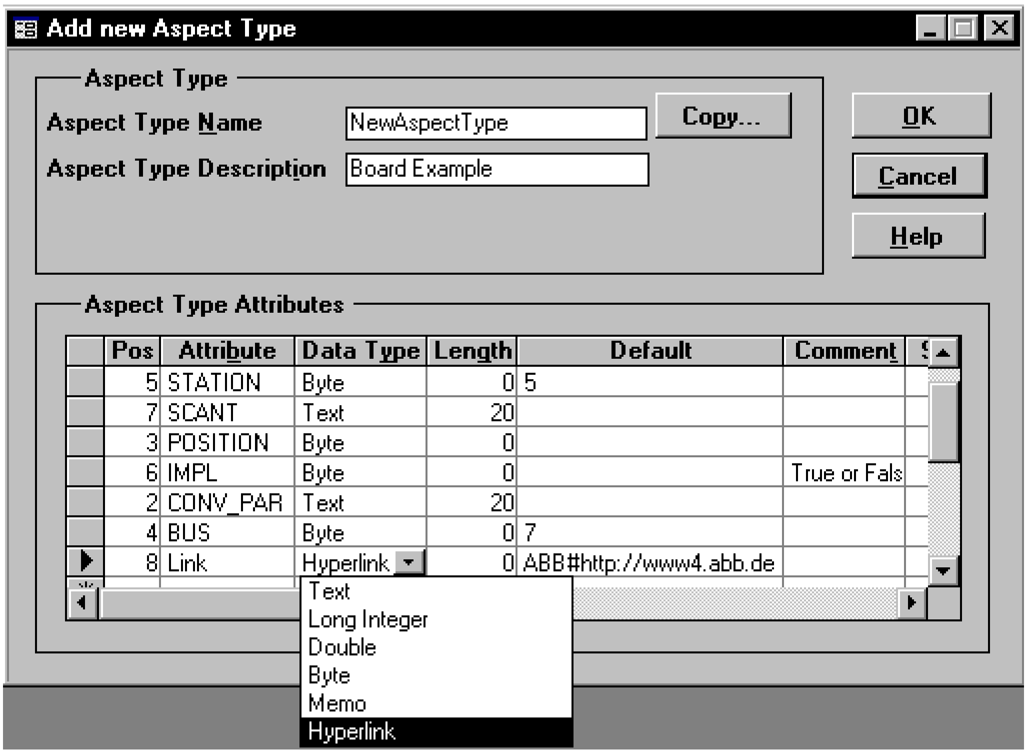
This dialog allows to enter Aspect Type Name, Aspect Type Description and Aspect Type Attributes. The fields **Aspect Type Name and Aspect Type Description** accept only letters, numbers and the underscore-sign as entry. After editing the Aspect Type name, the field **Aspect Type Description** offers the same string as default-value. It is possible to edit another Aspect Type description instead of the default-string.

New **Aspect Type Attributes** can be add in the table Aspect Type Attributes. It is necessary to select a data type from the list-field **Data Type**. The **Pos** column number defines the position of attributes in formsheet (for the 2 columns standard layout the form style follows the rule first down then over, see Figure 3-21).



Object Name: Example Project EXP12R0
Description: Basic Property
AspectDescription: AspectTypeA
AspectTypeName: AspectTypeA
Attribute1: 127
Attribute2: DI_110
Attribute3: 14
Record: 1

Figure 3-22. Example: Aspect Instance of "AspectTypeA"



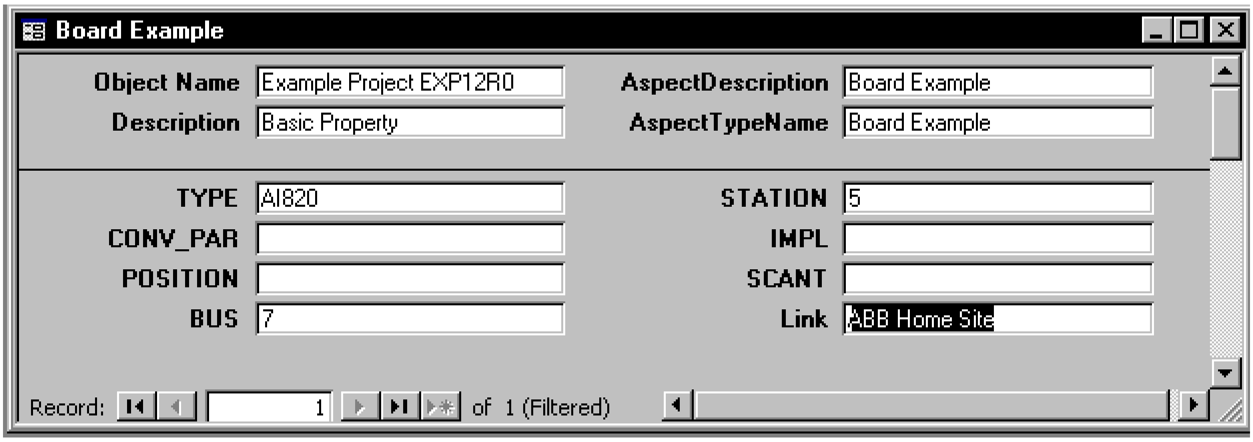
Aspect Type Name: NewAspectType
Aspect Type Description: Board Example

Pos	Attribute	Data Type	Length	Default	Comment
5	STATION	Byte	0	5	
7	SCANT	Text	20		
3	POSITION	Byte	0		
6	IMPL	Byte	0		True or Fals
2	CONV_PAR	Text	20		
4	BUS	Byte	0	7	
8	Link	Hyperlink	0	ABB#http://www4.abb.de	

Figure 3-23. Select the Attribute Data Type

Only the data type **Text** needs an entry in the field **Length**, it has to be in the range between 1 and 255. Hyperlink data type gives the opportunity to store a link to file or URL. **Default** field offers the possibility to enter default values which will be automatically inserted in the corresponding field at creation of the Parameter aspect instance (see Figure 3-22 and Figure 3-24). The default value has of course to fit in the data range of the selected data type. The syntax of Hyperlink data looks like:

- <Display text> **#** <Link>
  URL example: **ABB Home Site#http://www4.abb.de** respectively (see Figure 3-24).
  File example: **Product Guide#j:\documents\product.doc**
- The Display text is optional, if it is not entered, the ordinary link will be visible.
  URL example: either **www4.abb.de** or **http://www4.abb.de** is usable. Recognize that the hash sign (#) as separator between display text and link has to be omitted.



Board Example

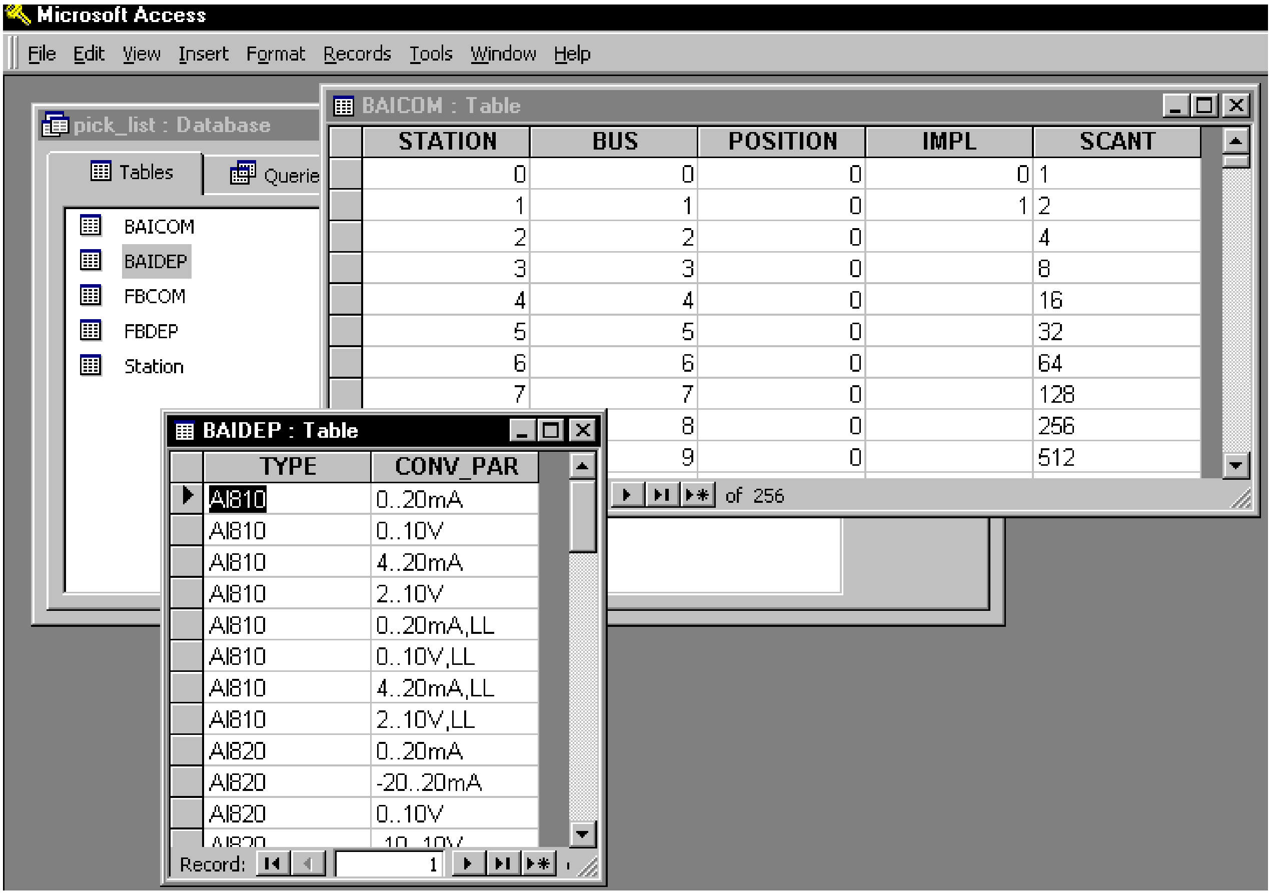
Object Name	Example Project EXP12R0	**AspectDescription**	Board Example
Description	Basic Property	**AspectTypeName**	Board Example
TYPE	AI820	**STATION**	5
CONV_PAR		**IMPL**	
POSITION		**SCANT**	
BUS	7	**Link**	ABB Home Site

Figure 3-24. Parameter Aspect with Default Values

Optionally you can use the field **Comment** for remarks. The entries in the comment field will automatically inserted into the Status Bar Help property and shown if the corresponded data field is touched by the user.

The fields **SourceType** and **Source** are prepared to offer the possibility to tie on user defined list of values so called pick-lists. The **SourceType** defines the type of pick-list, where to look up the values. The pick-list mechanism is optionally and you may use it for each Aspect Type Attribute. It will be possible both, to enter a list of values directly during creation or modification of an Aspect Type or to give the name of a table where to look up the pick-list values. It will also be possible to define a set of values for a Parameter attribute which depends from a value of a pick-list of one other attribute of the same Aspect Type. If the pick-list values will be supported via table, those tables have to be attached on the corresponding project database. The usable values for the Aspect Type Attributes who are supported by pick-lists are not limited on there attached list of values; the users are unrestrained to enter additional values too, there will be no check or validation except data type checks.

How to Prepare Pick-lists



BAICOM : Table

STATION	BUS	POSITION	IMPL	SCANT
0	0	0	0	1
1	1	0	1	2
2	2	0		4
3	3	0		8
4	4	0		16
5	5	0		32
6	6	0		64
7	7	0		128
	8	0		256
	9	0		512

BAIDEP : Table

TYPE	CONV_PAR
AI810	0..20mA
AI810	0..10V
AI810	4..20mA
AI810	2..10V
AI810	0..20mA,LL
AI810	0..10V,LL
AI810	4..20mA,LL
AI810	2..10V,LL
AI820	0..20mA
AI820	-20..20mA
AI820	0..10V

Figure 3-25. Example for User Defined Pick-lists

1. Create e.g inside Access database tables where the field names and data types fit to the attribute names and data types of the Parameter Aspect Type you want to support with a list of values and enter the relevant values. See Figure 3-25, table BACOM who covers values for the attributes: STATION, BUS, POSITION and so on.
2. If you want to use the “**Dependent on**” functionality prepare the table in a way as shown in Figure 3-25, see table BAIDEP where attribute CONV_PAR is dependent on the TYPE attribute.
3. Save your pick-list database somewhere in your project data area.

How to Tie on Pick-lists

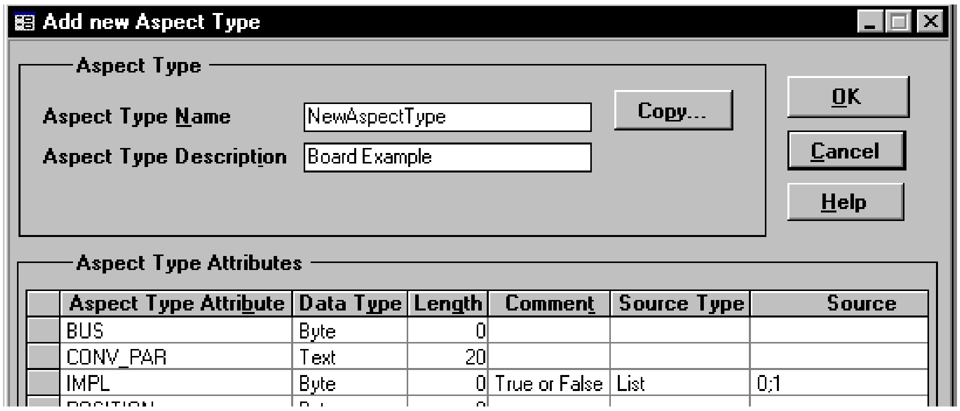
The following **Source Types** are prepared:

- List
- Table
- Dependent on.

Select List

(pick-list values stored in Source column)

Enter your values in the “**Source**” column separated by semicolon “;”.



Aspect Type Name: NewAspectType
Aspect Type Description: Board Example

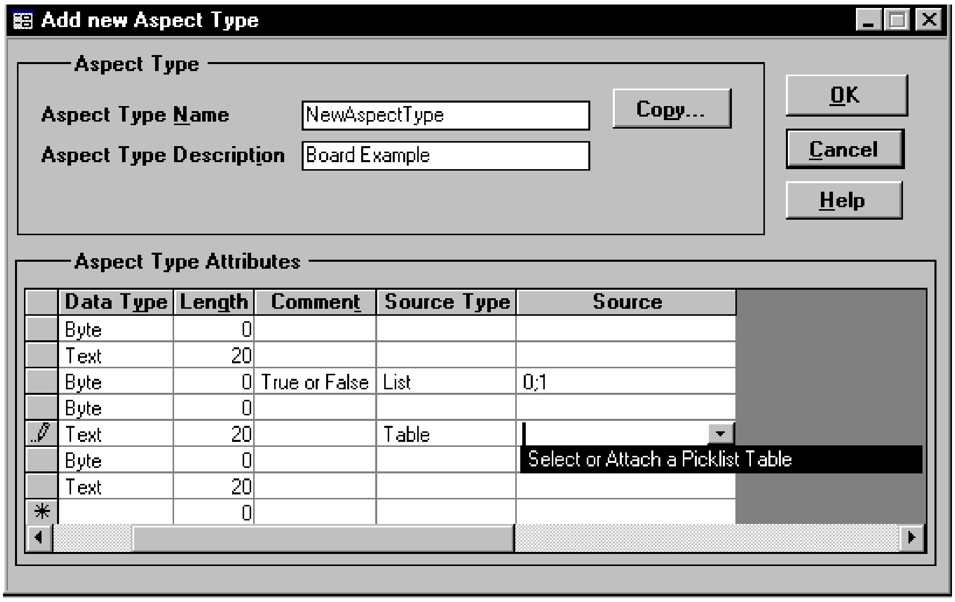
Aspect Type Attribute	Data Type	Length	Comment	Source Type	Source
BUS	Byte	0			
CONV_PAR	Text	20			
IMPL	Byte	0	True or False	List	0;1

Figure 3-26. Select Source Type List

Select Table

(pick-list values stored in an attached table)

1. Select “**Table”** in column **Source Type**.
2. Double click in column **Source** “Select or Attach a Picklist Table” .



Aspect Type Name: NewAspectType
Aspect Type Description: Board Example

	Data Type	Length	Comment	Source Type	Source
	Byte	0			
	Text	20			
	Byte	0	True or False	List	0;1
	Byte	0			
	Text	20		Table	
	Byte	0			
	Text	20			
*		0			

Figure 3-27. Select Source Type Table

3. Follow the dialogs to link your pick-list database and to attach your tables onto the project database.

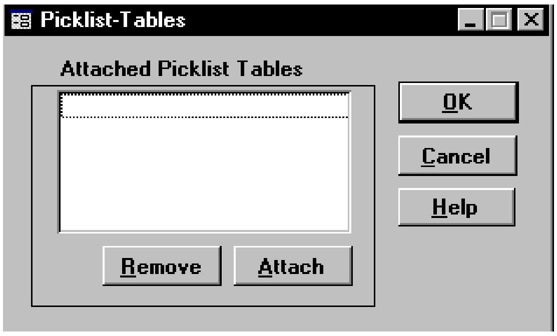
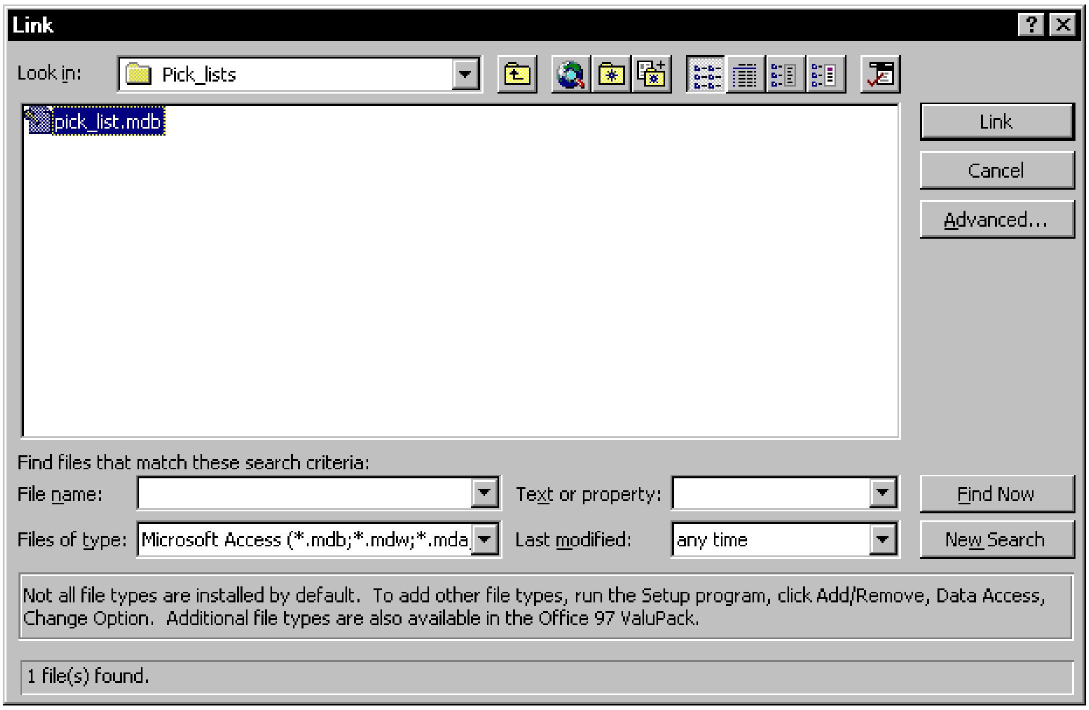


Figure 3-28. Attach Pick-list Dialog (Initial State)

4. Press **Attach** button to link your pick-list database onto your project database.



Look in: Pick_lists

pick_list.mdb

Files of type: Microsoft Access (*.mdb;*.mdw;*.mda

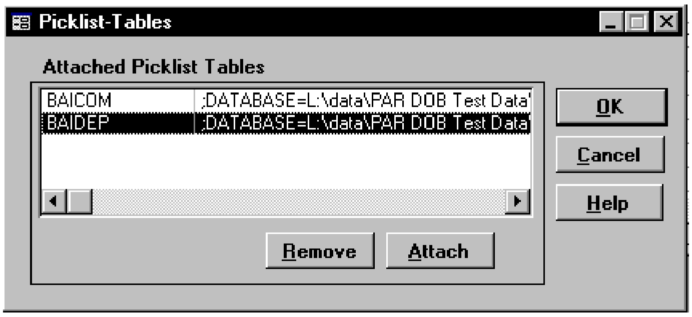
Last modified: any time

Not all file types are installed by default. To add other file types, run the Setup program, click Add/Remove, Data Access, Change Option. Additional file types are also available in the Office 97 ValuPack.

1 file(s) found.

Figure 3-29. Dialog to Link Pick-list Database on Project Database

5. Browse for pick-list database and press **Link** button.

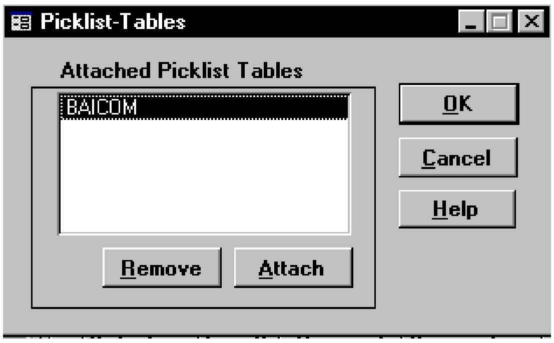


Attached Picklist Tables

BAICOM ;DATABASE=L:\data\PAR DOB Test Data'
BAIDEP ;DATABASE=L:\data\PAR DOB Test Data

Figure 3-30. Link Tables Dialog

6. Select the table you want to attache and press OK.

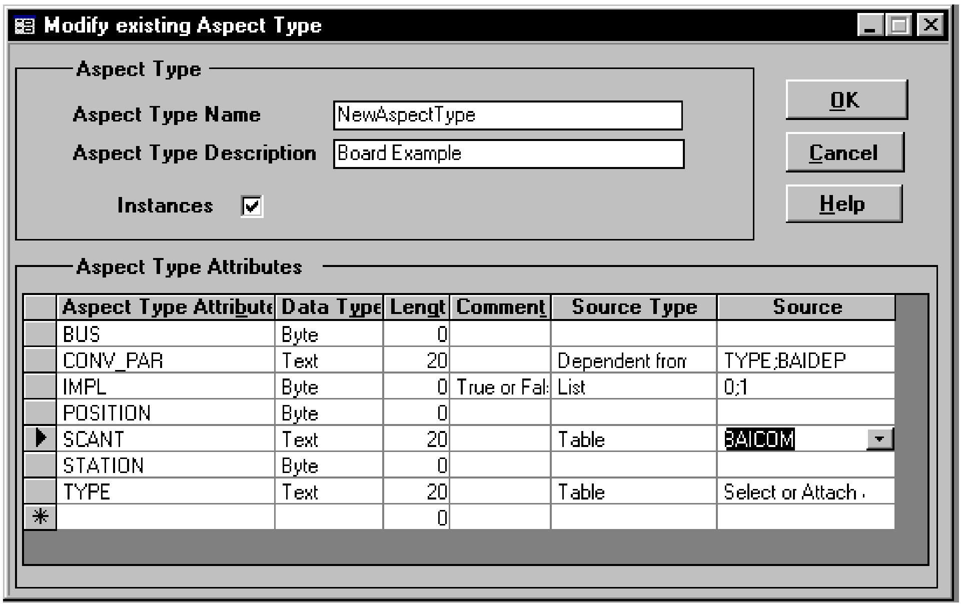


Attached Picklist Tables

BAICOM

Figure 3-31. Dialog to Attach a Table on an Aspect Attribute

7. Select the table who covers the column which serves the Aspect Attribute (SCANT) with values and press **OK.**



Aspect Type

Aspect Type Name: NewAspectType

Aspect Type Description: Board Example

Instances ☑

Aspect Type Attributes

	Aspect Type Attribut	Data Type	Lengt	Comment	Source Type	Source
	BUS	Byte	0			
	CONV_PAR	Text	20		Dependent fron	TYPE;BAIDEP
	IMPL	Byte	0	True or Fal	List	0;1
	POSITION	Byte	0			
▶	SCANT	Text	20		Table	BAICOM
	STATION	Byte	0			
	TYPE	Text	20		Table	Select or Attach .
*			0			

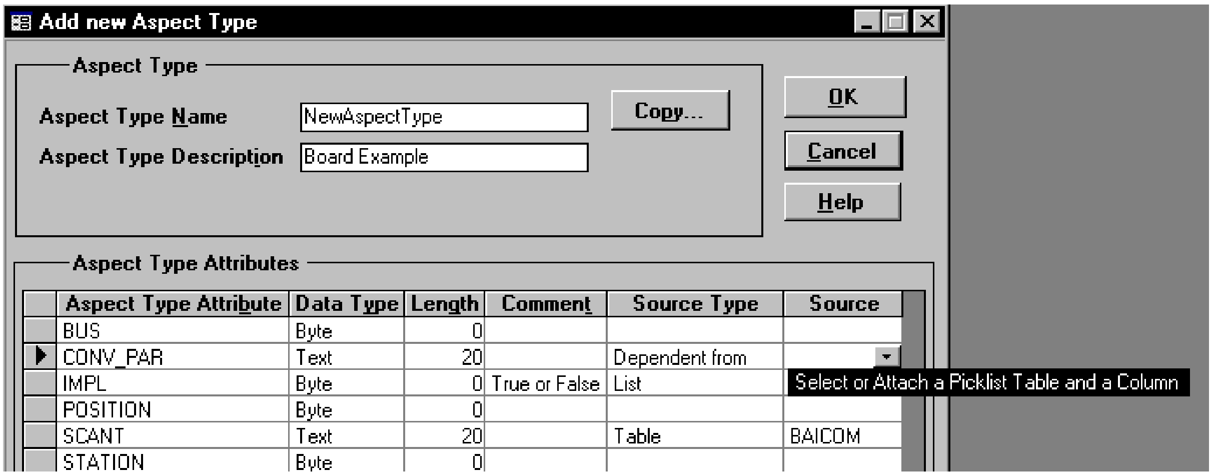
Figure 3-32. Aspect Type Attribute and Its Attached Table

The Aspect Type Attribute SCANT has got a pick-list where the values are stored in table BAICOM.

Select Dependent on

(pick-list values stored in an attached table and the dependent on attribute)

1. For the favored Aspect Type Attribute select "**Dependent on"** in column **Source Type**.
2. Double click in column **Source** "Select or Attach a Picklist Table and a Column".



Aspect Type Name: NewAspectType

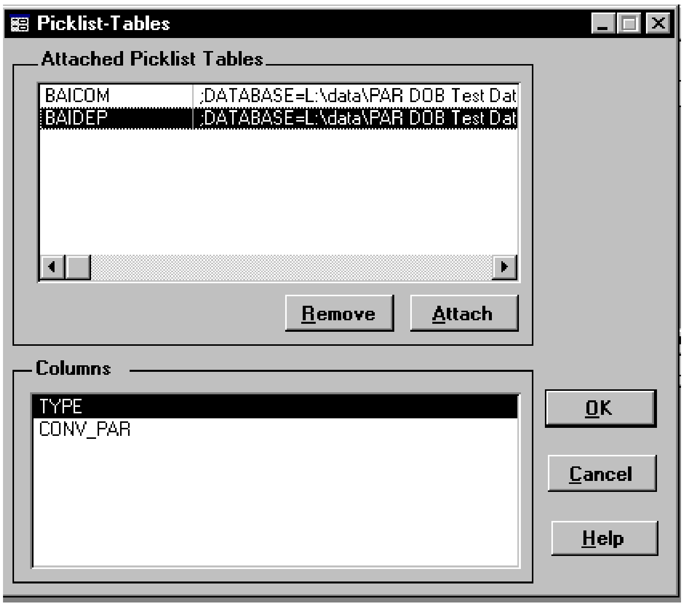
Aspect Type Description: Board Example

Aspect Type Attribute	Data Type	Length	Comment	Source Type	Source
BUS	Byte	0			
CONV_PAR	Text	20		Dependent from	
IMPL	Byte	0	True or False	List	
POSITION	Byte	0			
SCANT	Text	20		Table	BAICOM
STATION	Byte	0			

Select or Attach a Picklist Table and a Column

Figure 3-33. Select Source Type Dependent on

3. Follow the dialogs to link your pick-list database and to attach your tables onto the project database (proceed as described for Source Type Table, see Figure 3-29).



BAICOM	;DATABASE=L:\data\PAR DOB Test Dat
BAIDEP	;DATABASE=L:\data\PAR DOB Test Dat

TYPE
CONV_PAR

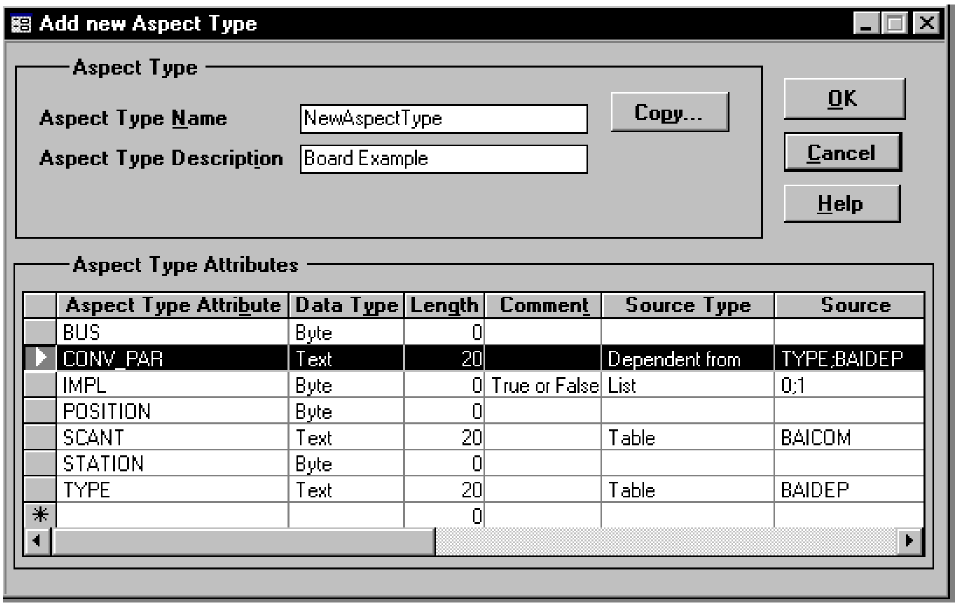
Figure 3-34. Dialog to Select Dependent on Attributes

The Aspect Type Attribute CONV_PAR pick-list values are dependent on TYPE values (see Figure 3-25).

4. Select pick-list table BAIDEP and column TYPE and press OK, Source field will be updated (see Figure 3-35).

NOTE

If you want to remove a table enter Picklist Table dialog select the corresponding table and press Remove button (see Figure 3-31 or Figure 3-34).

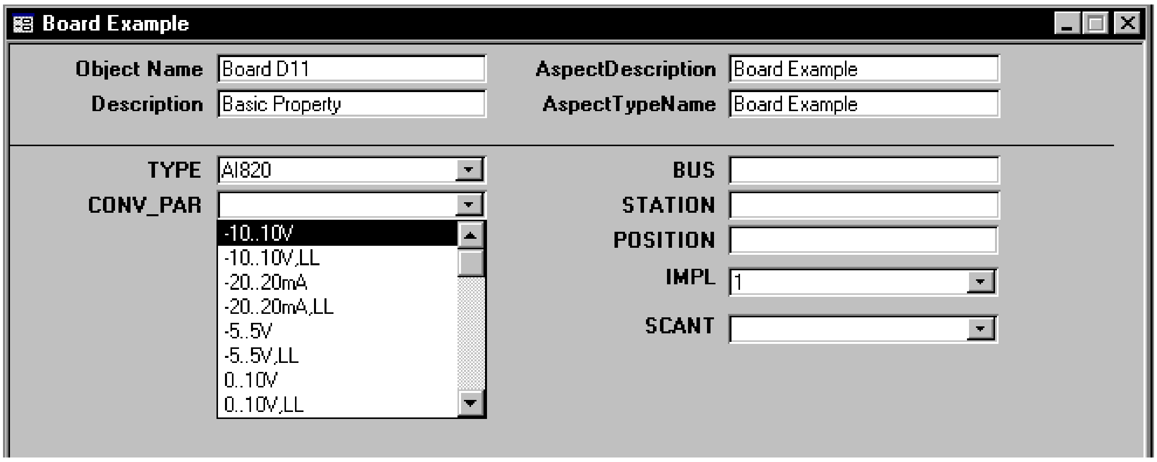


Aspect Type Name: NewAspectType
Aspect Type Description: Board Example

	Aspect Type Attribute	Data Type	Length	Comment	Source Type	Source
	BUS	Byte	0			
▶	CONV_PAR	Text	20		Dependent from	TYPE;BAIDEP
	IMPL	Byte	0	True or False	List	0;1
	POSITION	Byte	0			
	SCANT	Text	20		Table	BAICOM
	STATION	Byte	0			
	TYPE	Text	20		Table	BAIDEP
*			0			

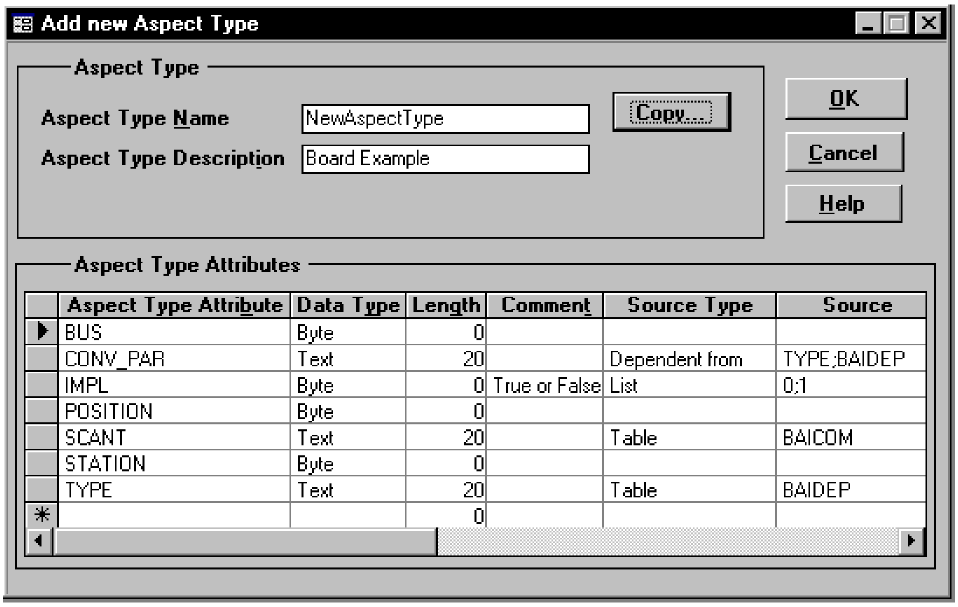
Figure 3-35. Aspect Type Attribute and Its Attached Table and Column

After creation of Parameter Aspect Type "NewAspectType" the generated template looks like:



Object Name: Board D11
Description: Basic Property
AspectDescription: Board Example
AspectTypeName: Board Example
TYPE: AI820
CONV_PAR:
BUS:
STATION:
POSITION:
IMPL: 1
SCANT:

Figure 3-36. NewAspectType Parameter Template: Formsheet View



Aspect Type Name: NewAspectType

Aspect Type Description: Board Example

Aspect Type Attribute	Data Type	Length	Comment	Source Type	Source
BUS	Byte	0			
CONV_PAR	Text	20		Dependent from	TYPE;BAIDEP
IMPL	Byte	0	True or False	List	0;1
POSITION	Byte	0			
SCANT	Text	20		Table	BAICOM
STATION	Byte	0			
TYPE	Text	20		Table	BAIDEP
		0			

Figure 3-37. Add new Aspect Type Buttons

The shown buttons and their operations:

- **OK** installs the new Aspect Type and closes the "Add new Aspect Type" dialog.
- **Cancel** aborts the procedure without saving any changes.
- **Help** shows the help-window of the Parameter Builder.
- **Copy...** allows to copy Aspect Types from the actual project or other projects. The button opens another window "Copy Aspect Type".

The **OK** button creates all Aspect Type entities (like queries, reports and templates) belonging to this Aspect Type (for example for the Aspect Type named = ProcessObjectParameter):

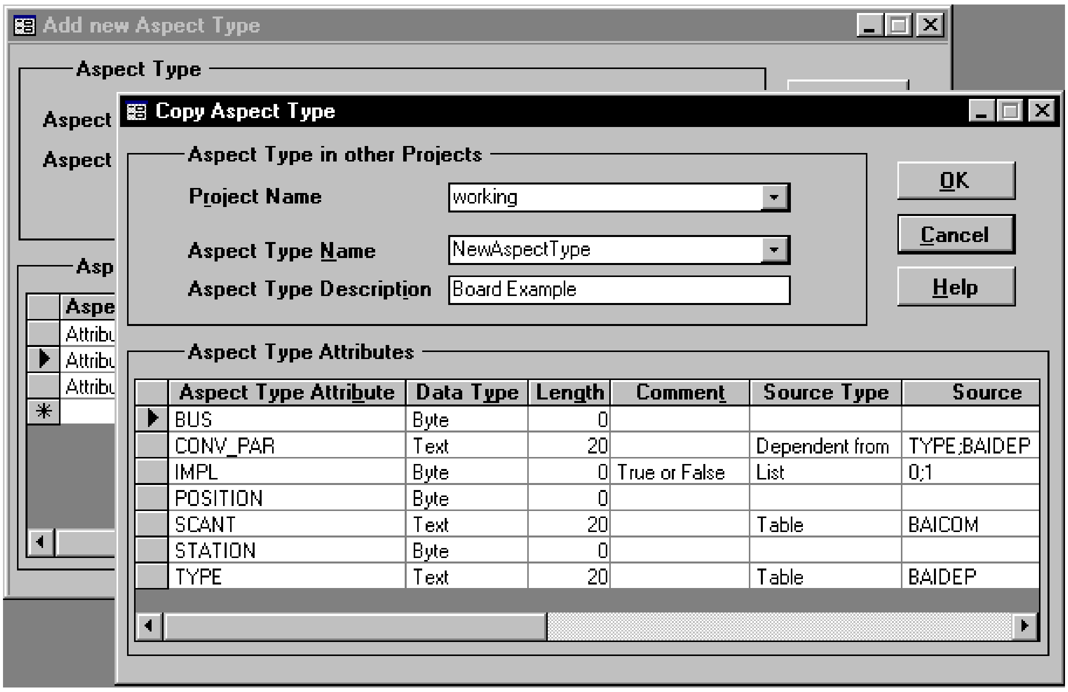
AspectTypeTable = ProcessObjectParameter
AspectTypeRelation = RProcessObjectParameter
AspectTypeQuery = QProcessObjectParameter
AspectTypeTemplate = FProcessObjectParameter
AspectTypeListReport = RProcessObjectParameterList
AspectTypeSingleReport = RProcessObjectParameterSingle

NOTE

The automatically created Aspect Type template and the Aspect Type reports are designed for a maximal length of about 15 characters as entry for Aspect Type Name, Aspect Type Description and Aspect Type Attributes.

Copy Aspect Type from the Actual Project or Other Projects

Instead of editing new entries for the new Aspect Type the **Copy...** Aspect Type button allows to copy the type definition out of any project.



Project Name: working

Aspect Type Name: NewAspectType

Aspect Type Description: Board Example

Aspect Type Attribute	Data Type	Length	Comment	Source Type	Source
BUS	Byte	0			
CONV_PAR	Text	20		Dependent from	TYPE;BAIDEP
IMPL	Byte	0	True or False	List	0;1
POSITION	Byte	0			
SCANT	Text	20		Table	BAICOM
STATION	Byte	0			
TYPE	Text	20		Table	BAIDEP

Figure 3-38. Copy Aspect Type

The dialog allows to copy Aspect Types with their Aspect Type Attributes. Aspect Type entities like templates, reports, etc. are **not copied**, but regenerated losing all user defined adjustments.

Select in the list-field **Project Name** a project as source to copy from. The field **Aspect Type Name** displays a list of Aspect Types found in the current selected project. The belonging **Aspect Type Description** and as well the **Aspect Type Attributes** are visible. After the copy is finished, the Aspect Type definitions can be adjusted and the Aspect Type can be renamed.

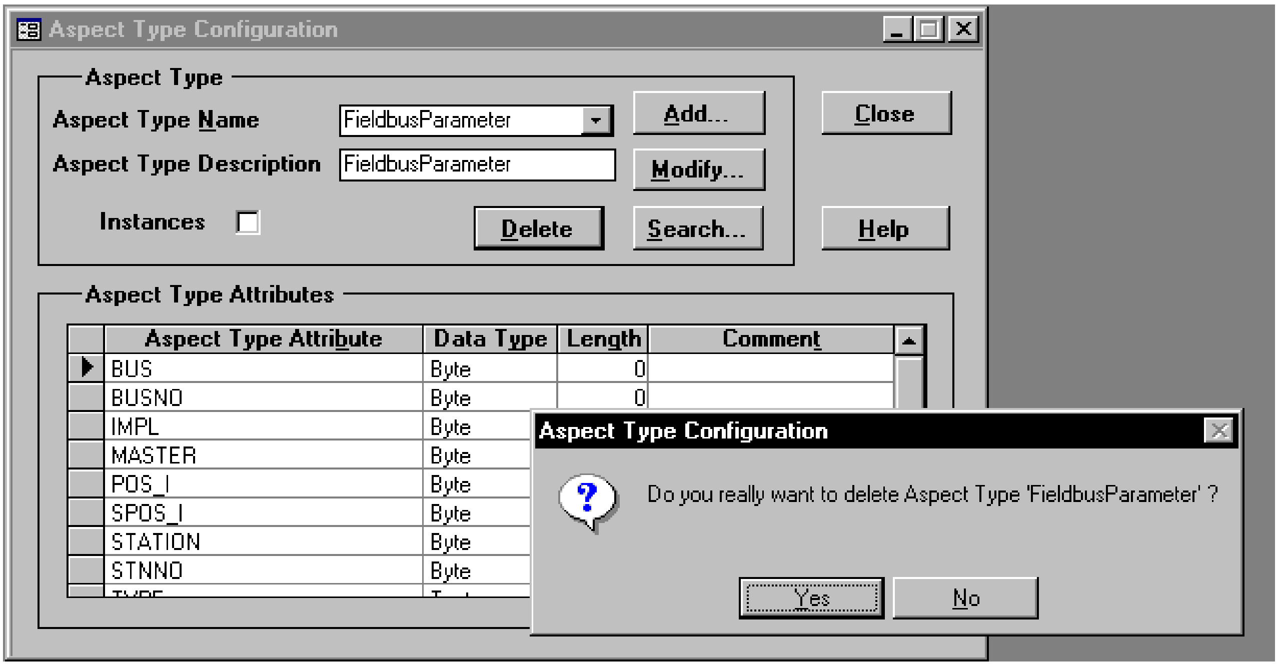
The shown buttons and their operations:

- **OK** inserts a copy of the specified Aspect Type into the **Add new Aspect Type** window and closes the **Copy Aspect Type** window.
- **Cancel** aborts the procedure and closes the window.
- **Help** shows the help-window of the Parameter Builder.

NOTE

On opening this dialog, all projects are scanned for Aspect Type definitions, so the amount of time for this action will increase with the number of projects found.

3.6.2.2 Delete Aspect Type



Aspect Type Name: FieldbusParameter
Aspect Type Description: FieldbusParameter
Instances ☐

Aspect Type Attribute	Data Type	Length	Comment
BUS	Byte	0	
BUSNO	Byte	0	
IMPL	Byte		
MASTER	Byte		
POS_I	Byte		
SPOS_I	Byte		
STATION	Byte		
STNNO	Byte		

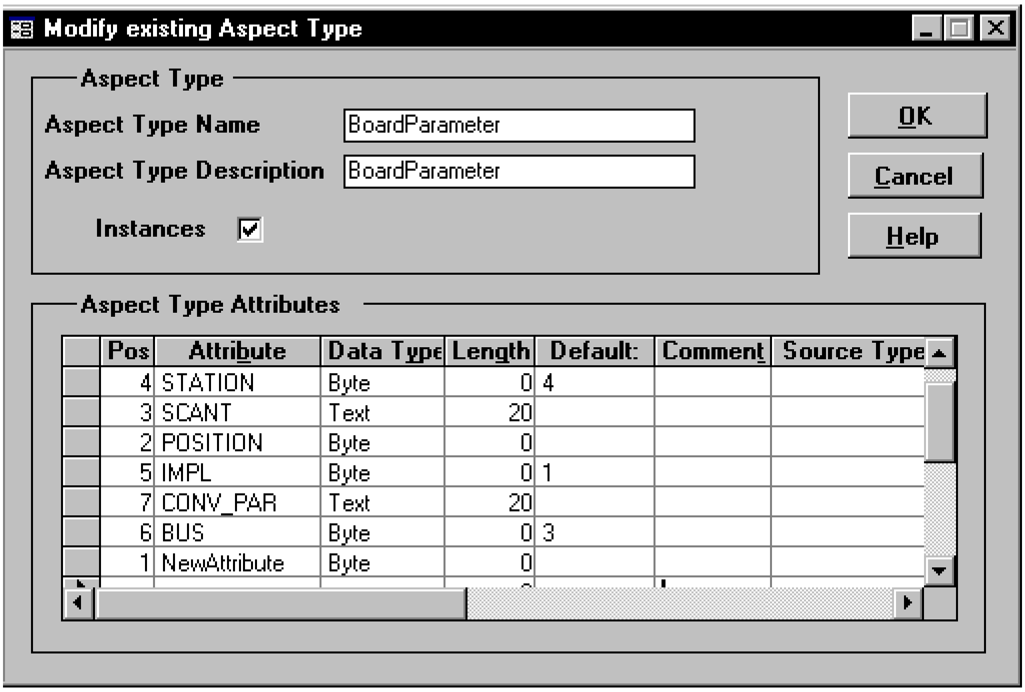
Do you really want to delete Aspect Type 'FieldbusParameter' ?

Figure 3-39. Delete Aspect Type

The **Delete** button allows to delete the current Aspect Type and the Aspect Type entities. This button is not visible, if the actual Aspect Type has instances.In this case delete all instances for this Aspect Type with the Structure Builder. Select by use of the Object Window of the Structure Builder the entire Advant objects which include Aspects of that Aspect Type and delete them one after another. Afterwards enter the **Aspect Type Configuration** dialog again and execute the deletion.

3.6.2.3 Modify Existing Aspect Type

This dialog allows to modify the actual Aspect Type.



Aspect Type Name: BoardParameter

Aspect Type Description: BoardParameter

Instances ☑

Pos	Attribute	Data Type	Length	Default:	Comment	Source Type
4	STATION	Byte	0	4		
3	SCANT	Text	20			
2	POSITION	Byte	0			
5	IMPL	Byte	0	1		
7	CONV_PAR	Text	20			
6	BUS	Byte	0	3		
1	NewAttribute	Byte	0			

Figure 3-40. Modify Aspect Type

NOTE

It is not possible to change the Aspect Type Name or the Aspect Type Description of an existing Aspect Type.

The name of Aspect Type Attributes can only be changed, if the actual Aspect Type has no instances (indicated by the **Instances** check box). Changing the **Data Type** might cause loss of data. In case of **Data Type** changes a message box informs the user about this risk. The **Pos** number offers the possibility to rearrange the attributes in formsheet.

NOTE

If **Pos** number was changed, in the **Keep Forms and Reports** dialog the **Forms Template** entry has not be checked (see Figure 3-41) otherwise the attribute rearrangement forced by **Pos** numbers will be dropped.

To connect pick-lists on existing Aspect Type Attributes proceed as described in Chapter 3.6.2.1.

Data Type Converting Rules:

Starting Data Type	Final Data Type	Result	Remark
Text	Byte	Data will be converted into the appropriate value	Text like “0”, ..,“255” otherwise none.
Text (n: characters)	Text (m)	Data will be converted	m >= n otherwise data will be truncated
Byte, Long Integer, Double	Text (m)	Data will be converted	m has to fit to capture the value otherwise number will be truncated
Byte	Double, Long Integer	Data will be converted	
Double, Long Integer	Byte	Data will be converted	If 0 <= data < 256 and none fractions
Double (fractions)	Long Integer	Data will be rejected	Only fractions will be rejected (“Type mismatch”)

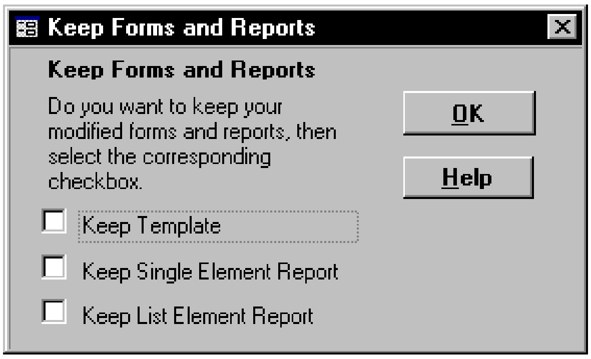
New Aspect Type Attributes can be added in the table “AspectTypeAttribute” without any restrictions.

The shown buttons and their operations:

- **OK** reinstalls the Aspect Type and closes the **Modify existing Aspect Type** dialog
- **Cancel** aborts the procedure without saving any changes
- **Help** shows the help-window of the Parameter Builder.

Reuse of Older Forms & Reports After Aspect Type Modification

The Aspect Type Wizard saves the templates and reports of an Aspect Type before modification. During the regeneration of the Aspect Type a question dialog is raised and the user can select the saved templates and reports to be used in the regeneration of the Aspect Type.



Keep Forms and Reports

Do you want to keep your modified forms and reports, then select the corresponding checkbox.

☐ Keep Template

☐ Keep Single Element Report

☐ Keep List Element Report

Figure 3-41. Keep Form and Reports Check Box

Enabling this feature preservers all user made modification in templates and reports, but the user has to adapt the templates and reports to the new aspect type definition manually. All Users that do not want to make the modifications manually, do not check the check boxes in the question dialog to modify and regenerate the templates and reports automatically.

This facility should only be used by users who are familiar with the form and report design in Microsoft Access, since manual modifications have to be done with the forms- and reports designer tools of Microsoft Access.
Using these tools, enables the user to change the layout of the templates and reports and the appearance of attribute-fields easily. Removed attributes can be deleted from the templates and reports and new attributes can be inserted. To insert a new attribute the attribute list window has to be opened and the desired attributes has to be dragged from the list into the template or report.

NOTE

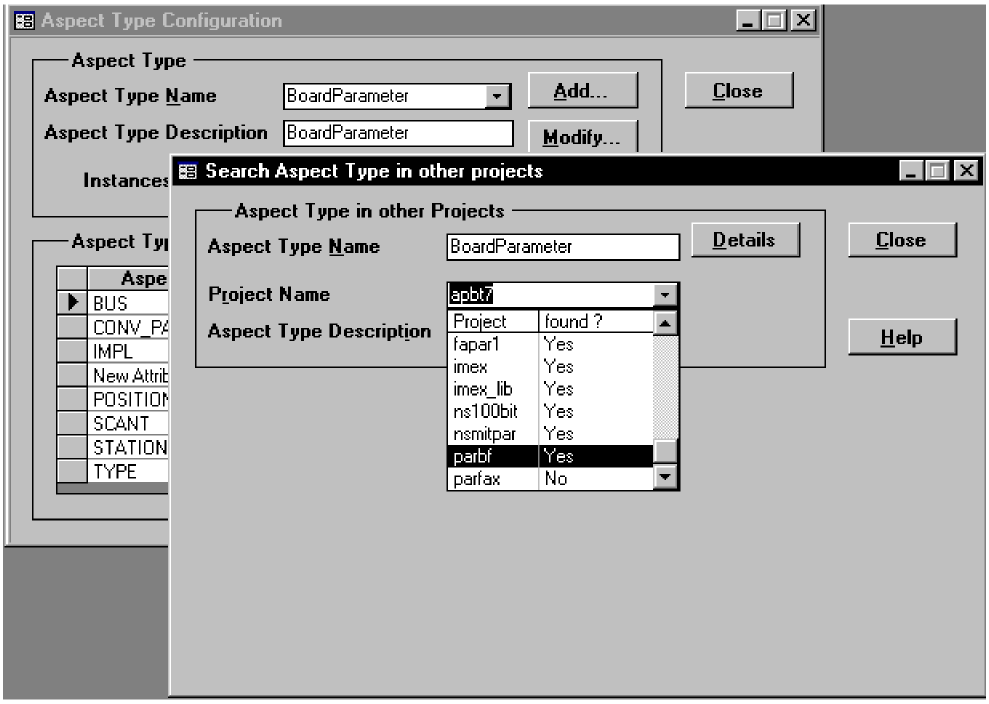
If you tied up a pick-list on an Aspect Type Attribute of **Source Type** “Dependent on” it will not be possible to keep the previous template therefor avoid to check “Keep Template”.

3.6.2.4 Search for Aspect Type in Other Projects

With this dialog it is possible to select a project by using the list-field **Project Name**. The list-field contains a remark “**found ? = Yes**”, if the belonging project contains the searched **Aspect Type Name**. The field **Aspect Type Description** displays the description of the Aspect Type in the selected project.

NOTE

If you activate the search dialog, all projects are scanned for Aspect Type definitions, so the amount of time for this action will increase with the number of projects found.



Aspect Type Name: BoardParameter

Aspect Type Description: BoardParameter

Search Aspect Type in other projects

Aspect Type Name: BoardParameter

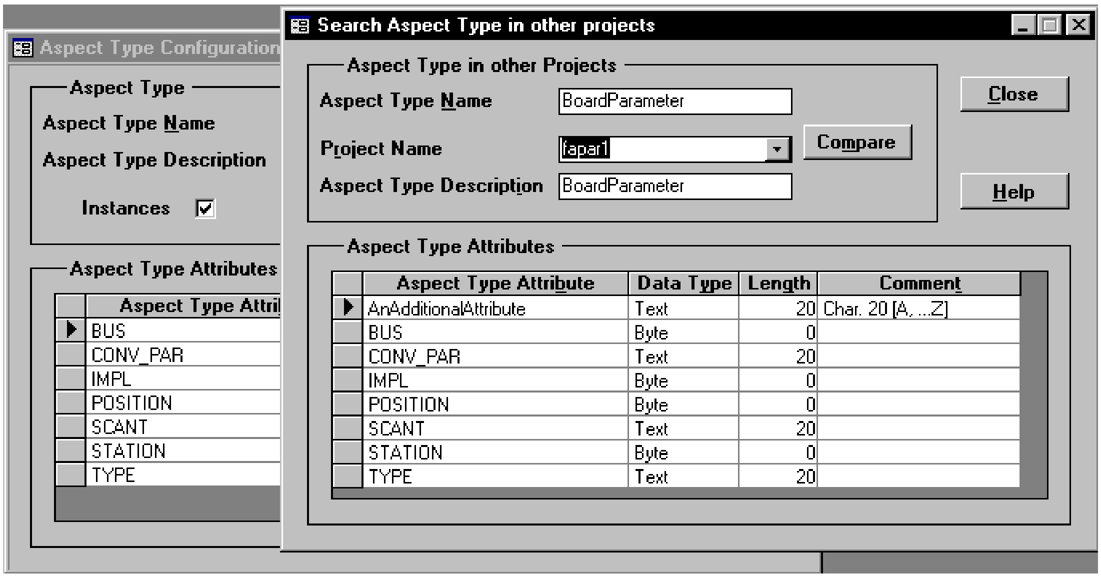
Project Name: apbt7

Aspect Type Description

Project	found ?
fapar1	Yes
imex	Yes
imex_lib	Yes
ns100bit	Yes
nsmitpar	Yes
parbf	Yes
parfax	No

Figure 3-42. Search for Aspect Type

The **Details** button shows the Aspect Type Attributes of the selected Aspect Type in the selected project and the additional button **Compare** comes into sight.



Aspect Type Name: BoardParameter

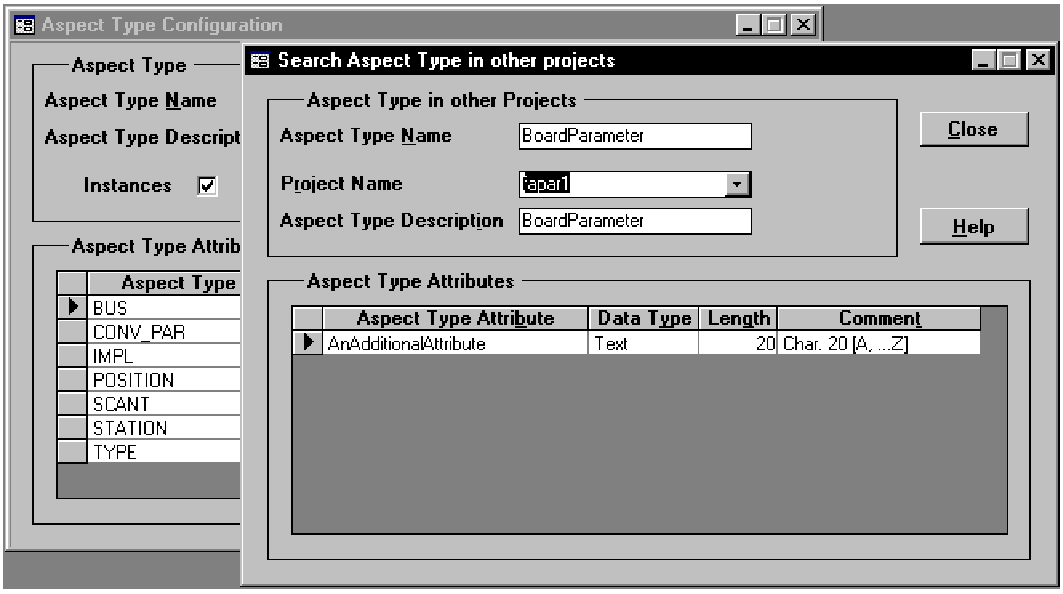
Project Name: fapar1

Aspect Type Description: BoardParameter

Aspect Type Attribute	Data Type	Length	Comment
AnAdditionalAttribute	Text	20	Char. 20 [A, ...Z]
BUS	Byte	0	
CONV_PAR	Text	20	
IMPL	Byte	0	
POSITION	Byte	0	
SCANT	Text	20	
STATION	Byte	0	
TYPE	Text	20	

Figure 3-43. Details of the Selected Aspect Type

After execute **Compare** only the Attribute-record sets with differences to the current project are selected.



Aspect Type
BUS
CONV_PAR
IMPL
POSITION
SCANT
STATION
TYPE

Aspect Type Name: BoardParameter
Project Name: tapar1
Aspect Type Description: BoardParameter

Aspect Type Attribute	Data Type	Length	Comment
AnAdditionalAttribute	Text	20	Char. 20 [A, ...Z]

Figure 3-44. Compare Aspect Types

3.6.2.5 Export Aspect Type Definition

In coherence to the **Aspect Type Configuration** dialog, the menu **File**, menu item **Export Aspect Type Definition** starts creating an ASCII-file with the naming rule:
project name.**CSV** in the directory, *advabld***data***proj-name***par\db**
that contains all Aspect Types and their related Attributes of the selected project. The entries of this file are comma delimited and for example ready to import into the Instrument Index Data Acquisition tool.

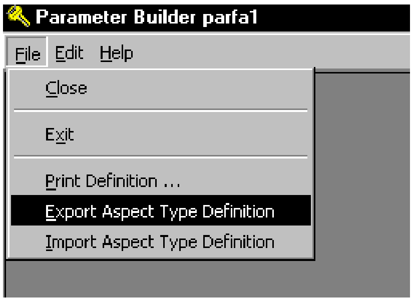


Figure 3-45. Export Aspect Type Definitions

As example the contents of a CSV-file (comma delimited) with the Aspect Type **NewAspectType** after execution of **Export Aspect Type Definition**:

Aspect Type Definition Structure

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment	SourceType	Source
NewAspectType	TYPE	Text	20		Table	BAIDEP
NewAspectType	CONV_PAR	Text	20		Dependent on	TYPE;BAIDEP
NewAspectType	SCANT	Text	20		Table	BAICOM
NewAspectType	IMPL	Byte	0	True or False	List	0;1
NewAspectType	BUS	Byte	0			
NewAspectType	STATION	Byte	0			
NewAspectType	POSITION	Byte	0			

3.6.2.6 Import Aspect Type Definition

Another menu item, **Import Aspect Type Definition**, allows to insert Aspect Types from the ASCII-file *projectname*.CSV to the **Aspect Type Configuration** dialog.

Prepare an ASCII file with the layout as shown above - see Aspect Type Definition File -and place it in the surrounding of the project where the Aspect Type should be inserted: *advabld***data***proj-name***par****db**

The import function searches only for new Aspect Types. Already existing Aspect Types can not be inserted once again. A message box informs the user, if a new Aspect Type is found in the CSV-file. The Message box allows to decline the insertion. In the course of the insertion the Aspect Type entities will be installed automatically. The Aspect Type Description in the **Aspect Type Configuration** dialog gets the identical entry as the Aspect Type Name.

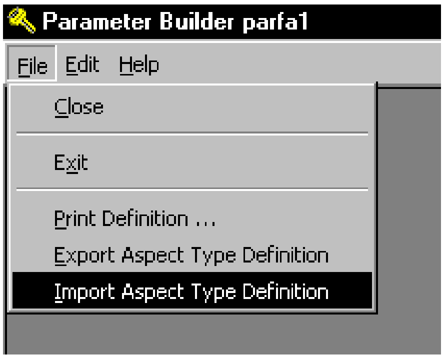


Figure 3-46. Import Aspect Type Definitions

3.6.2.7 Print Aspect Type Definition

The menu item dialog **Definition** opens a report representing a list of all current defined Aspect Types and Aspect Type Attributes.

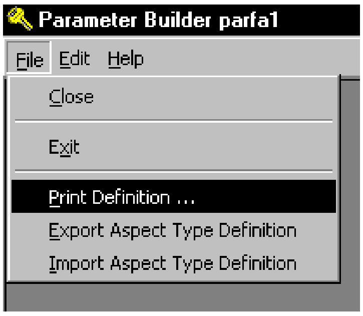
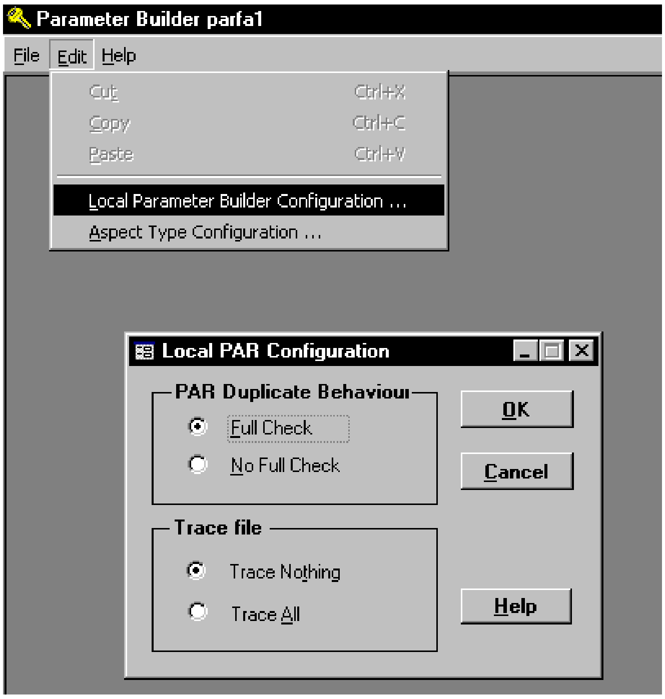


Figure 3-47. Print Definition

3.6.3 Local Parameter Builder Configuration

The menu "**Edit**" contains the menu item **Local PAR Configuration...** to configure the logfile-handling of the Parameter Builder. Start-up the belonging dialog **Local PAR Configuration**, it displays the current configuration for **PAR Duplicate Behaviour** and **Trace file**; those settings are stored in the ***par.ini***- file.



PAR Duplicate Behaviour
● Full Check
○ No Full Check

Trace file
● Trace Nothing
○ Trace All

Figure 3-48. Local Parameter Builder Configuration

NOTE

To put your changes in operation close project and open it again.

The shown buttons and their operations.

- After selecting a new configuration **OK** saves the entries and closes the current window.
- **Cancel** allows to leave the window without saving the configuration.
- **Help** shows the help-window of the Parameter Builder.

Parameter Builder Duplicate Behavior

The success of the duplicate method depends on the compatibility of the source parameter aspect types (for example in the library project) and the aspect types of the actual project. In principle it is not possible to instantiate Advant objects with attached parameter aspects out of a library project into the actual project where the parameter aspect types do not match.

The Parameter Builder duplicate configuration behavior opens the opportunity either to follow the above mentioned method in a way that

- when the duplicate method is set to "**Full Check**" then parameter aspect instances can only be duplicated (instantiate) between projects, if their aspect types are fully compatible.

or

- when the duplicate method is set to "**No Full Check**" then parameter aspect instances will be duplicated even if the aspect types do not fit.

NOTE

Not fit means, none of the source attributes are identical to the actual attributes, see also below.

Example: Duplicate behavior

Duplicate behavior Set to	**Attributes of the source Parameter Aspect Type**	**Attributes of the actual Parameter Aspect Type**	**Instantiation/ Copy**
No Full Check	*A1, B1, C1*	*A1, B1, C1*	**yes**
No Full Check	*A1, B1, C1*	*A1, B1*	**no**
No Full Check	*A1, B1, C1*	*U1, V1, W1*	**yes but source data is not copied!**
No Full Check	*A1, B1, C1*	*A1, U1, V1, W1*	**no**
No Full Check	*A1, B1, C1*	*A1, B1, C1, U1, V1, W1*	**yes**
Full Check	*A1, B1, C1*	*A1, B1, C1*	**yes**
Full Check	*A1, B1, C1*	*U1, V1, W1*	**no**
Full Check	*A1, B1, C1*	*A1, U1, V1, W1*	**no**
Full Check	*A1, B1, C1*	*A1, B1, C1, U1, V1, W1*	**yes**

At instantiation through an import, the parameter aspect data of the source project will be copied into the parameter aspect of the actual project, but afterwards overridden with the parameter aspect data from the import file. In this case the parameter aspect instance in the source project serves as a template. If, for example the aspect data of the source aspect is not needed, the Duplicate method can be set to "**No Full Check**", which means, if it is not possible to instantiate an parameter aspect instance from the source and copy the attribute data because parameter aspect types are incompatible, the desired parameter aspect instance is created in the actual project without data. The attributes will be filled out later in the import process. It could also be useful while copying Advant objects from a library manually.

Trace File

Selecting the "**Trace All**" option activates process tracing in to trace file ***par.log*** on temporary directory (usually) ***c:\tmp***. The trace file contains information about internal processing of the parameter builder, which is helpful to analyze failure situations. It should be disabled by default. The trace file is not cumulative, it is deleted on each start of the parameter builder.If the environment variable **"TMP"** is set, the logfile is created in the location the variable points to.

3.6.4 Configuration of Templates and Reports

The build in Aspect Types of the Parameter Builder are delivered with manually created templates and reports. All user created Aspect Types will get automatically generated templates and reports. For each Aspect Type a template and two reports are created (see Section 3.6.2.1, Add New Aspect Type).

The automatic template and report generation simply arranges the data fields on a default template or report. The user can modify these templates and reports to adapt them to his needs.Reports can be redefined by the user by following the steps:

1. Open a Parameter Aspect of the desired Aspect Type from the Structure Builder.
2. Choose the datasheet view of the Aspect Type template, if the list-report should be modified.
3. Perform the menu entry **Print Preview** from the **File** menu.
4. Open the report design mode by pressing the **Design View** button.
5. Use the standard Microsoft Access Report Designer to customize the report.

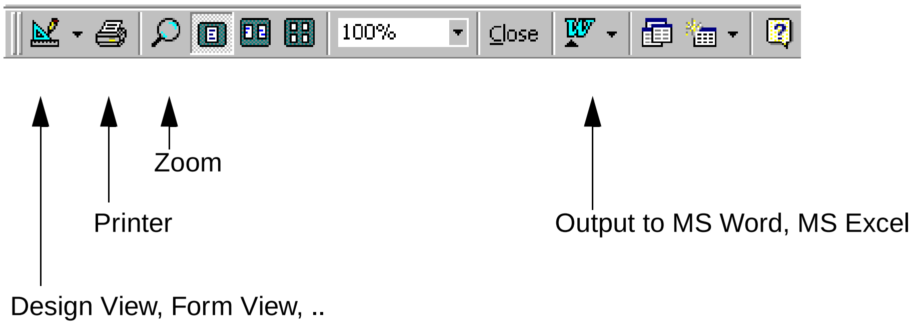


Figure 3-49. Toolbar Print Preview

Templates can be redefined by the user by following the steps:

1. Open a Parameter Aspect of the desired Aspect Type from the Structure Builder.
2. Open the template in design mode by pressing the **Design View** button.
3. Use the standard Microsoft Access Forms Designer to customize the template

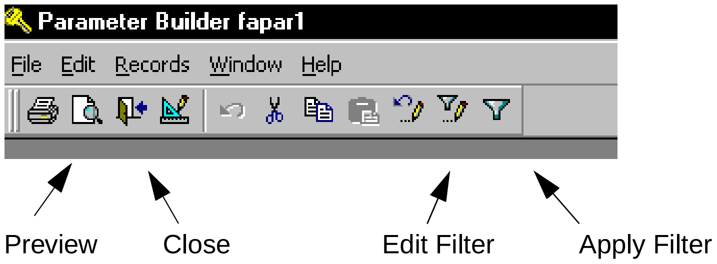


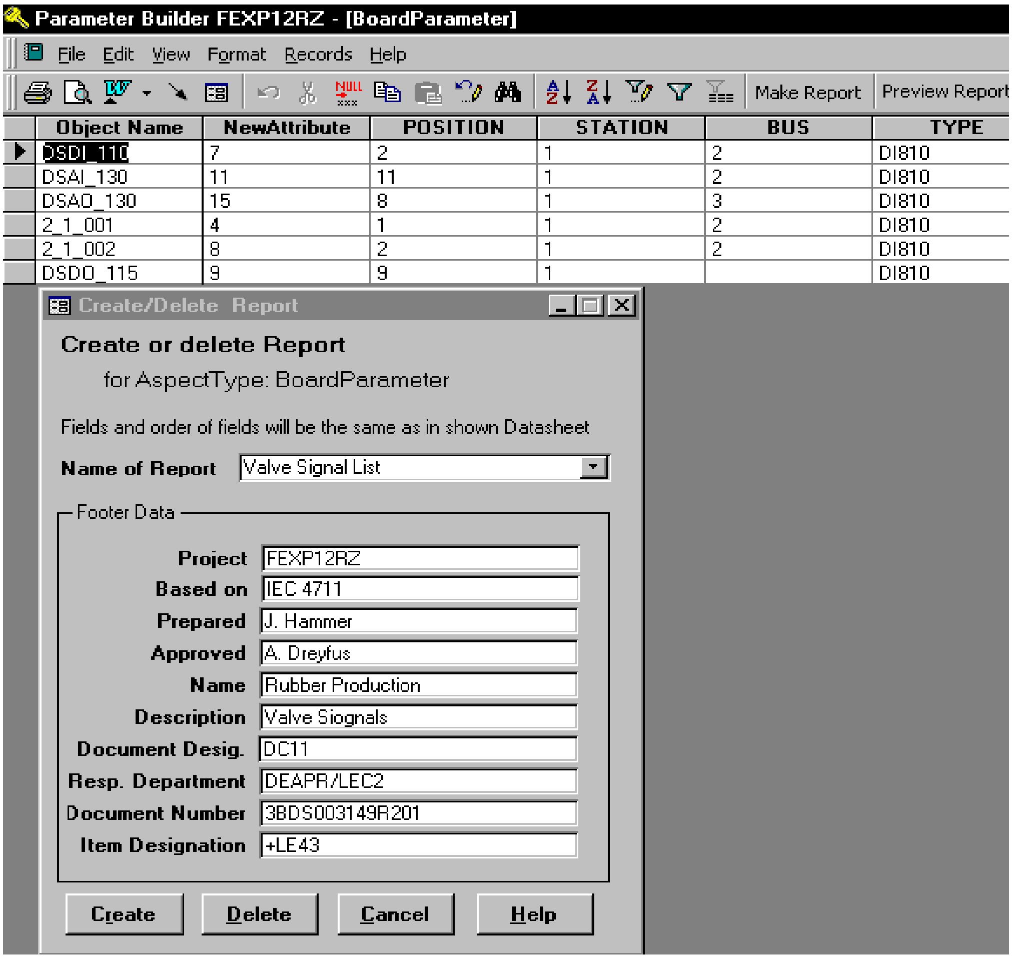
Figure 3-50. Toolbar for Formsheet View of Aspect Type Templates

The actual configuration of templates and reports is performed using MS Access standard functions (please refer to the user manual of MS Access). The help button in the design toolbar leads to the Microsoft Access help instead of Parameter Builder help.

Remark: The templates and reports contain fields, that are not visible in formsheet- or datasheet view. Do not touch these fields, otherwise the template (form) or report will not work correctly.Modifications to form or report headers, footers and properties can effect functionality. For details see Section 3.4.3, Parameter Builder (Parameter Template).

3.6.5 User Defined Reports

At aspect type configuration PAR prepares for every aspect type a standard list report with the whole set of attributes and a predefined fields layout.


Parameter Builder FEXP12RZ - [BoardParameter]
File Edit View Format Records Help
Make Report
Preview Report
Create/Delete Report
Create
Delete
Cancel
Help

Object Name	NewAttribute	POSITION	STATION	BUS	TYPE
DSDI_110	7	2	1	2	DI810
DSAI_130	11	11	1	2	DI810
DSAO_130	15	8	1	3	DI810
2_1_001	4	1	1	2	DI810
2_1_002	8	2	1	2	DI810
DSDO_115	9	9	1		DI810

Create or delete Report

for AspectType: BoardParameter

Fields and order of fields will be the same as in shown Datasheet

Name of Report Valve Signal List

Footer Data

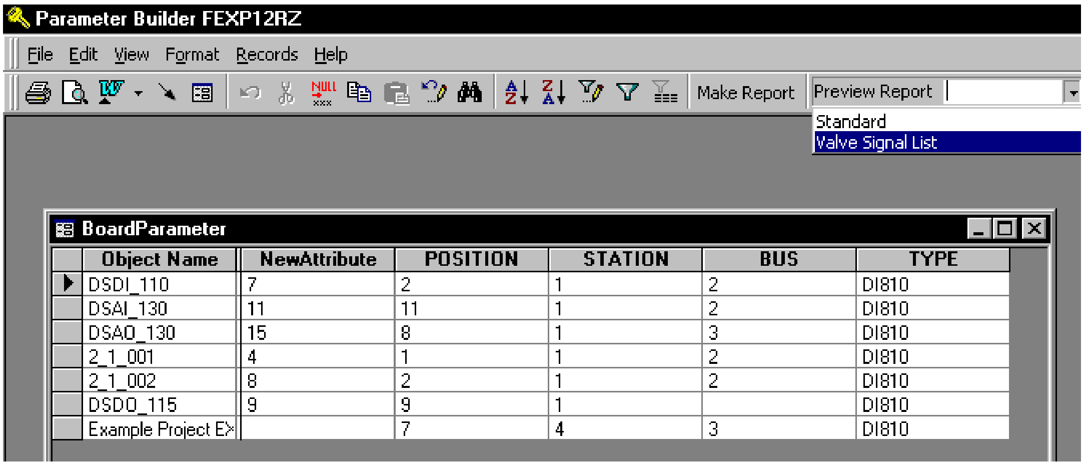
Project	FEXP12RZ
Based on	IEC 4711
Prepared	J. Hammer
Approved	A. Dreyfus
Name	Rubber Production
Description	Valve Siognals
Document Desig.	DC11
Resp. Department	DEAPR/LEC2
Document Number	3BDS003149R201
Item Designation	+LE43

Figure 3-51. Create Delete Report Dialog

The Make Report button in toolbar offers the possibility to define a report based on the just active datasheet with the shown columns and the selected aspect instances. To setup the proper datasheet proceed in that way:

1. Menu **Format**, menu item **Unhide Columns..** allows to select the set of columns which should be shown in report.
2. The order of fields presented in report have to be arrange by drag and drop of the columns in datasheet form.
3. To get a subset of aspect data define a filter (see Figure 3-9).
4. Click **Make Report** button in toolbar.
5. Enter report name and footer data.
6. Make report by clicking **Create** button.

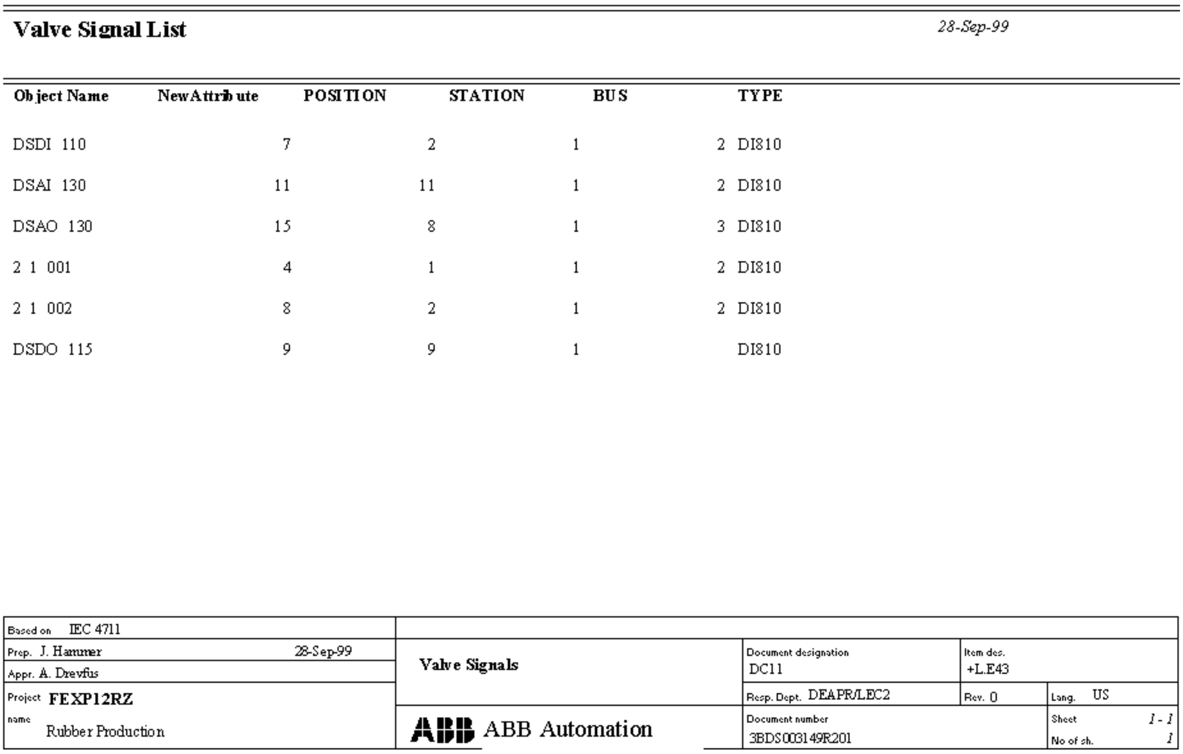
To preview and to print report click **Preview Report** button in toolbar and pick out the wanted report. The list of presented reports depends on aspect types where the actual datasheet belongs to.



Object Name	NewAttribute	POSITION	STATION	BUS	TYPE
DSDI_110	7	2	1	2	DI810
DSAI_130	11	11	1	2	DI810
DSAO_130	15	8	1	3	DI810
2_1_001	4	1	1	2	DI810
2_1_002	8	2	1	2	DI810
DSDO_115	9	9	1		DI810
Example Project E		7	4	3	DI810

Figure 3-52. Preview Report Button

Print preview toolbar respectively menu **File** presents the possibility to print report or among other things to output report to Word or Excel.



Valve Signal List

28-Sep-99

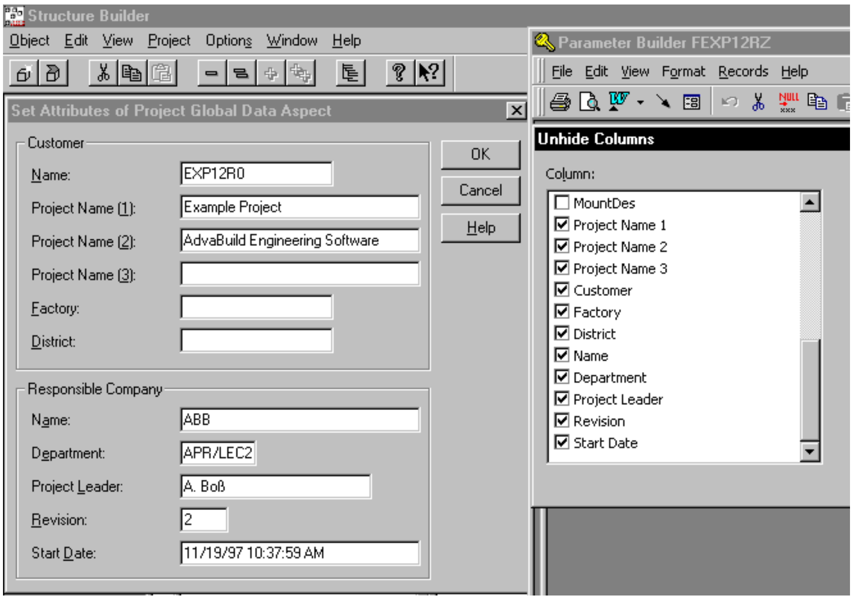
Object Name	NewAttribute	POSITION	STATION	BUS	TYPE
DSDI 110	7	2	1	2	DI810
DSAI 130	11	11	1	2	DI810
DSAO 130	15	8	1	3	DI810
2 1 001	4	1	1	2	DI810
2 1 002	8	2	1	2	DI810
DSDO 115	9	9	1		DI810

Based on IEC 4711				
Prep. J. Hammer 28-Sep-99	Valve Signals	Document designation DC11	Item des. +L.E43	
Appr. A. Drevfus				
Project FEXP12RZ		Resp. Dept. DEAPR/LEC2	Rev. 0	Lang. US
name Rubber Production	ABB Automation	Document number 3BDS003149R201		Sheet 1 - 1 No of sh. 1

Figure 3-53. User Defined Report

3.6.5.1 Reuse Project Global Data

The fields and data of the Structure Builder Project Global Data aspect are integrated in the different Parameter aspect type datasheet forms. As default behavior those fields are hidden. Make them visible in datasheet by clicking menu **Format**, item **Unhide Columns..**.


Structure Builder
Object Edit View Project Options Window Help
Parameter Builder FEXP12RZ
File Edit View Format Records Help
OK
Cancel
Help

Set Attributes of Project Global Data Aspect

Customer

Name: EXP12R0
Project Name (1): Example Project
Project Name (2): AdvaBuild Engineering Software
Project Name (3):
Factory:
District:

Responsible Company

Name: ABB
Department: APR/LEC2
Project Leader: A. Boß
Revision: 2
Start Date: 11/19/97 10:37:59 AM

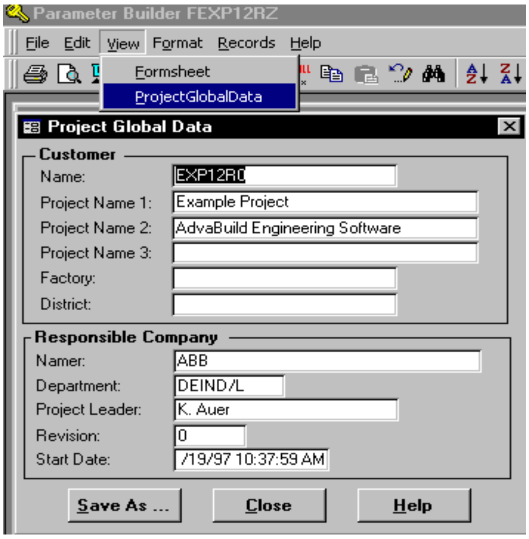
Unhide Columns

Column:

☐ MountDes
☑ Project Name 1
☑ Project Name 2
☑ Project Name 3
☑ Customer
☑ Factory
☑ District
☑ Name
☑ Department
☑ Project Leader
☑ Revision
☑ Start Date

Figure 3-54. Reuse Project Global Data Fields in Parameter Datasheet

Beside the opportunity to use project global data in reports it is also possible to export data to Word, Excel or ordinary text file for further interpretation by user.


Parameter Builder FEXP12RZ
File Edit View Format Records Help
Formsheet
ProjectGlobalData
Save As ...
Close
Help

Project Global Data

Customer

Name: EXP12R0
Project Name 1: Example Project
Project Name 2: AdvaBuild Engineering Software
Project Name 3:
Factory:
District:

Responsible Company

Name: ABB
Department: DEIND/L
Project Leader: K. Auer
Revision: 0
Start Date: /19/97 10:37:59 AM

Figure 3-55. Project Global Data Form

Start up **Project Global Data** view by clicking menu **View**, item **ProjectGlobalData.** Through button **Save as..** filename, location and file type will be asked, default is L:\<project name>\par\projectglobaldata.xls.

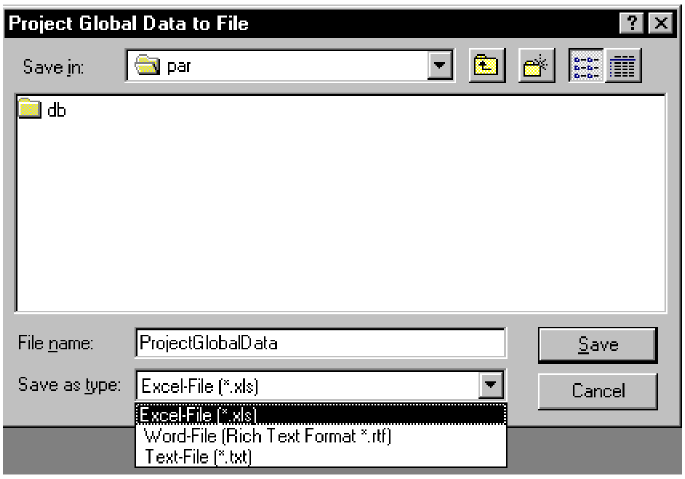


Figure 3-56. Project Global Data Save as Dialog

3.7 Configuration/Application Building Menus and Toolbars

3.7.1 Menus

The following commands are accessible in the menus of the

- Formsheet of a Parameter Template
- Datasheet of a Parameter Template
- Filter Dialog
- Print Preview Dialog
- Report Designer Dialog
- Forms Designer Dialog.

3.7.1.1 Formsheet

File Edit Records Window Help

File

- Close
  Close the form but keep the Parameter Builder application active (see also Section 3.4.3.5, Close the Parameter Template).
- Save Record
  Save the modifications done in the data base.
- Print Preview
  Opens a report in the preview mode. The report displays attributes of the actual parameter aspect.
- Print
  Prints the report showing attributes of the actual parameter aspect.
- Exit
  Terminates the Parameter Builder application (see also Section 3.4.3.5, Close the Parameter Template).
-

Edit

- Undo typing or Redo (u) Typing respectively
  Undo or redo the latest typing in a field.
- Undo Current Field or Undo Current Record respectively
  Undo the latest modifications in a field. If several fields have been modified only the latest field modification can be undone. After that the menu entry changes to “Undo Current Record” which allows to undo all modifications of the record (provided it is not yet saved).
- Cut
  Cuts a string selected with the mouse.

- Copy
  Copies a string selected with the mouse.
- Paste
  Paste the string cut or copied before.
- Local Parameter Builder Configuration
  Opens a dialog to view/change local settings.
- Aspect Type Configuration
  Opens a dialog to administrate aspect types.
- Filter by Form
  Enter filter conditions direct in form.

Records

- Edit Filter/Sort...
  Opens the Edit Filter dialog (see Section 3.4.3.5, Close the Parameter Template).
- Apply Filter/Sort
  Performs the Filter edited before (see also Section 3.4.3.5, Close the Parameter Template).

Window

- Tile Horizontally
- Tile Vertically
- Cascade.

Help

- Contents
  Opens Parameter Builder Help.
- Search
  Opens Parameter Builder Help with search dialog.
- Access Contents and Index
- How to use Help
  Opens Parameter Builder Help.
- Release Notes
  Opens the release notes document.
- Show Project History
  Project life cycle information and version.
- Check Installation
  AEWcheck utility starts up to evaluate the AEW environment.
- About Parameter Builder
  Displays the actual version of the Parameter Builder.

3.7.1.2 Datasheet

File

Same as in the formsheet .

Edit

- Undo typing or Redo (u) Typing respectively
  Undo or redo the latest typing in a field.
- Undo Current Field or Undo Current Record respectively
  Undo the latest modifications in a field. If several fields have been modified only the latest field modification can be undone. After that the menu entry changes to “Undo Current Record” which allows to undo all modifications of the record (provided it is not yet saved).
- Cut
  Cuts a string selected with the mouse.
- Copy
  Copies a string selected with the mouse.
- Paste
  Paste the string cut or copied before.
- Find
  Searches for a certain string in one or several attributes of one or several parameter aspects (see also Section 3.4.3.2, Search for Parameter Values).
- Replace
  Replaces a string in one or several attributes of one or several parameter aspects (see also Section 3.4.3.3, Modify Attributes).
- Replace NULL-Values
  Substitute NULL values by given value for selected column.

View

- Formsheet
  Switches back to the formsheet.
- ProjectGlobalData
  Form to show and export project global data.

Format

- Font...
  Font settings for the values displayed.
- Row Height...
  Settings for row dimensions.
- Column Width...
  Settings for columns dimensions.
- Unhide Columns
  Hide or unhide columns on the screen.

- Freeze Columns
  Fix certain columns that they are not scrolled out of the window.
- Unfreeze Columns
  Unfreeze all columns frozen before.

Records

- Go To
  Jump to a certain row in the datasheet.
- Refresh
  Refresh the contents of the datasheet.
- Synchronize
  Synchronize object name and absolute reference designations.
- Quick Sort
  Sorting of a column either ascending or descending (see also Section 3.4.3.2, Search for Parameter Values).
- Edit Filter/Sort...
  Same as in the formsheet view.
- Apply Filter/Sort
  Same as in the formsheet view.
- Filter by Form
  Enter filter conditions direct in form.
- Show All Records
  If a filter has been applied only a subset of the parameter aspects are displayed. This menu entry allows you to show all parameter records.

Help

Same as in the formsheet .

3.7.1.3 Filter Dialog

The filter dialog is an MS Access standard dialog. However some of the entries of the standard menu of MS Access have been removed. The remaining entries are:

File

- Close
  Close the filter dialog.
- Save As Query
  Save a query (filter settings) for later reuse.
- Load From Query
  Load a query (filter settings) saved before in order to apply it.

Edit

- Undo
  Same as in formsheet.
- Cut
  Same as in formsheet.
- Copy
  Same as in formsheet.
- Paste
  Same as in formsheet.
- Delete
- Clear Grid
  Removes the gridlines from the filter dialog.
- Insert Row
  Inserts a new row in order to add an "OR-related" filter criteria.
- Delete Row
  Remove an "OR-related" filter criteria.
- Insert Column
  Inserts a new column in order to add an "AND-related" filter criteria.
- Delete Column
  Remove an "AND-related" filter criteria.

Records

Apply Filter/Sort
Apply the filter.

Help

About Parameter Builder
Same as in formsheet.

3.7.1.4 Print Preview Dialog

The print preview dialog is an MS Access standard dialog. However some of the entries of the standard menu of MS Access have been removed. The remaining entries are:

File View Help

File

- Close
  Close dialog.
- Output To
  save report contents to an external file in one of the following formats:
  - MS Excel

  - Rich Text Format
  - MS DOS Text
  - or extra Database respectively within the current database.
- Print Setup
  Print configurations.
- Print
  Print the report.

View

Design
Opens Report Designer.

Help

About Parameter Builder
Same as in formsheet.

3.7.1.5 Report Designer Dialog

The report design dialog is an MS Access standard dialog. However some unnecessary entries of the standard menu of MS Access have been removed.

3.7.1.6 Forms Designer Dialog

The forms design dialog is an MS Access standard dialog. However some unnecessary entries of the standard menu of MS Access have been removed.

3.7.2 Toolbars

The following commands are accessible in the toolbar of the

- Formsheet of a Parameter Template
- Datasheet of a Parameter Template
- Filter Dialog
- Print Preview Dialog
- Report Designer Dialog.

3.7.2.1 Formsheet

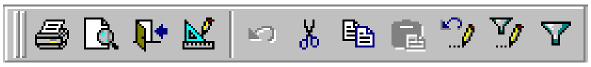


Figure 3-57. Formsheet Toolbar

The buttons are addressed from left to right:

- Print
  same as **Print** in the **File** menu
- Preview
  same as **Print Preview** in the **File** menu
- Close
  same as **Close** in the **File** menu
- Design
  opens Forms Designer
- Undo
  same as **Undo** in the **Edit** menu
- Cut
  same as **Cut** in the **Edit** menu
- Copy
  same as **Copy** in the **Edit** menu
- Paste
  same as **Paste** in the **Edit** menu
- Undo Current Field/Record
  same as **Undo Current Field/Record** in the **Edit** menu
- Edit Filter
  same as **Edit Filter** in the **Record** menu
- Apply Filter
  same as **Apply Filter** in the **Record** menu

3.7.2.2 Datasheet

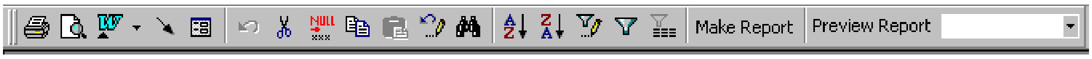


Figure 3-58. Datasheet Toolbar

The buttons perform the same functions as in the formsheet view (see above). The following buttons (addressed from left to right) are added in the datasheet view:

- Formsheet
  Switch to formsheet view.
- Find
  Finds a certain string.
- Sort
  Quick sort of the data of the current attribute (field) either ascending or descending. The current field is the one the cursor is located in.
- Show All Records
  Ignore the filter settings and show all records (parameter aspects).

3.7.2.3 Print Preview Dialog

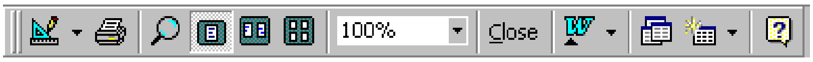


Figure 3-59. Print Preview Toolbar

- Switch to report designer, Form View, Datasheet View.
- Print
  Same as **Print** in the **File** menu.
- Print Setup
  Same as **Print Setup** in the **File** menu.
- Zoom
  Zoom in / out of the report.
- Show one, two or multiple pages
- Zoom, Page width
- Close
  Same as **Close** in the **File** menu.
- Output Data to Word, Excel
- Database window
- Create new object (Database, Form etc.)
- MS Access Help

3.7.2.4 Report Design Dialog

The toolbar of the report design dialog is an MS Access standard toolbar. However some unnecessary buttons of the standard toolbar of MS Access have been removed.

3.7.2.5 Forms Design Dialog

The toolbar of the report design dialog is an MS Access standard toolbar. However some unnecessary buttons of the standard toolbar of MS Access have been removed.

Chapter 4 Runtime Operation

4.1 Operating Overview

Not applicable.

4.2 Runtime Tutorial

Not applicable.

4.3 Operating Instructions

Not applicable.

4.4 Runtime Operations Menus

Not applicable.

Chapter 5 Maintenance

5.1 Error Messages

5.1.1 Database Error Messages

Table 5-1. Error Messages of Database Application

Error Number	Error Text
1	“An sever error occurred, message is :”
2	“An error occurred, message is :”
3	“Warning, message is :”
12	Error accessing file :
13	Error accessing DDE-command parameter file :
14	Imported data belongs to different target tables
15	Maximum number of errors reached :
16	“Error : project database version is incompatible with library database version, terminating application”
17	“Import cannot be performed, concurrent action is running on database”
18	Cannot evaluate automatic import error table
19	Cannot analyze mapping information
100	A Database error occurred message is :
101	Query does not retrieve any Records
102	Error executing SQL-statement
103	Database inconsistency detected object is :
104	No data to process
105	Error during insert or update item is :
106	Data integrity error in ImportRelations on filename :
107	Input Query not found :
108	A Query has to be specified

Table 5-1. Error Messages of Database Application (Continued)

Error Number	**Error Text**
109	A File has to be specified
110	"Object is not consistently inserted into database, skipped"
111	"Errors occurred during import of data, contents of error table see below:"
112	Mapping Query is erroneous
113	"Temporary table still exists, dropping table :"
114	Dropping table failed
115	"Import failed, temporary table not created :"
116	"Creation of mapping-information file failed, error was :"
117	"Unsupported aspect type found in mapping file, ignored :"
118	There are no Parameter Builder aspect types contained in the mapping file
119	Found no attributes for aspect type :
120	No Data found to write to mapping info file
121	Cannot access logbook table
122	Cannot access import data table
1000	Error during clean up after processing
1001	Action cancelled by user
1002	processing aborted !
1003	current processing skipped !
1004	Import Errors -*
2001	Initializing
2002	Reading Command/Configuration Files
2003	Processing Mapfile
2004	Import Data
2005	Analyze Data
2006	Creating Queries
2007	Working
2008	Cleanup
2009	Complete data
2010	Processing Datafile

Table 5-1. Error Messages of Database Application (Continued)

Error Number	Error Text
2011	Processing temporary data table
2012	Checking imported Data
2013	Update Data Tables
5000	Logfilename is :
5001	Datafilename is :
5002	Query applied is :
5003	Rows retrieved from query :
5004	Rows imported from file
5005	Actual processed object is :
5006	Columns retrieved from query
5007	Columns imported from file
5008	It's not possible to delete data in this form! Please delete corresponding aspect in the Structure Builder.
5009	It's not possible to insert data in this form! Please insert corresponding aspect in the Structure Builder.
5010	Do you really want to cancel the import process ?
5011	Import processing stopped at object :
5012	Update statement is:
5013	Mapfile name is :
5014	Mapping file does not contain any aspect types belonging to Parameter Builder
5999	
8100	Local PAR Configuration
8110	Invalid entry for 'PAR Duplicate Behaviour' found in PAR.INI !
8120	Invalid entries for 'Trace file' found in PAR.INI !
8130	No option selected for 'PAR Duplicate Behaviour' !
8140	No option selected for 'Trace file' !
8199	
8200	Aspect Type Configuration
8201	Export Aspect Type Definition

Table 5-1. Error Messages of Database Application (Continued)

Error Number	Error Text
8202	Import Aspect Type Definition
8210	The field AspectTypeName needs an entry !
8211	The field AspectTypeDescription needs an entry !
8212	The field AspectTypeAttribute needs an entry !
8220	Unknown error occurred deleting Query 'Q<AspectTypeName>' !
8221	Unknown error occurred deleting Relation 'R<AspectTypeName>' !
8222	"Unknown error occurred deleting Table '<AspectTypeName>' !
8223	Unknown error occurred deleting Form 'F<AspectTypeName>' !
8224	Unknown error occurred deleting Report 'R<AspectTypeName>Single' !
8225	Unknown error occurred deleting Report 'R<AspectTypeName>List' !
8230	Unknown error occurred checking existence of Table '<AspectTypeName>' !
8231	Unknown error occurred creating Table '<AspectTypeName>' !
8232	Unknown error occurred creating Entries for AspectType '<AspectTypeName>' !
8233	Unknown error occurred creating Relation 'R<AspectTypeName>' !
8234	Unknown error occurred creating Query 'Q<AspectTypeName>' !
8235	Unknown error occurred creating Form 'F<AspectTypeName>' !
8236	Unknown error occurred creating Reports 'R<AspectTypeName>***' !
8240	Do you really want to delete Aspect Type '<AspectTypeName>' ?
8250	Import is active ! Aspect Type Configuration is therefore not allowed !
8260	Aspect Type Definition successfully exported to file <filename> !
8270	Import Aspect Type '<AspectTypeName>' from file '<FileName>' !
8272	No new Aspect Types found in file '<FileName>' !
8299	
8400	Aspect Add / Modify Configuration
8401	Add new Aspect Type
8402	Modify existing Aspect Type
8410	The specified AspectTypeName '<AspectTypeName>' already exists !

Table 5-1. Error Messages of Database Application (Continued)

Error Number	Error Text
8411	The specified AspectTypeDescription ‘<AspectTypeDescription>’ already exists !
8420	New Length < old Length ! Instances will be lost !
8422	DataType ‘<OldDataType>’ changed to ‘<NewDataType>’ !
8424	Instances will be lost !
8426	Instances can be assumed !
8430	The String ‘<TempName>’ is too long ! (maximal 100 characters !)
8432	The String ‘<TempName>’ contains the invalid character ‘<Character>’ !
8434	The AspectTypeAttribute<AspectTypeAttribute>’ is too long ! (maximal 50 characters !)
8436	The AspectTypeAttribute’<AspectTypeAttribute>’ contains the invalid character ‘<Character>’ !
8438	The specified AspectTypeAttribute’<AspectTypeAttribute>’ already exists !
8440	Length must be between 1 and 255 for DataType ‘Text’ !
8450	Do you really want to quit ?
8499	
8600	Aspect Search / Copy Configuration
8601	Search Aspect Type in other projects
8602	Copy Aspect Type from other projects
8610	AspectType ‘<AspectTypeName>’ not found in project ‘<ProjectName>’
8620	Parameter-DataBase not correct installed for project ‘<ProjectName>’ !
8630	No other projects found !
8699	
9999	“This is a dummy number, if it is found, no error message has to be retrieved”

5.2 Reporting Problems

Please report any problem by filling in a Product Maintenance Report (PMR).

Moreover in case of additional information is required please provide a copy of the logfile which is stored in C:\tmp\par.log by default.

5.3 What to Do at Program Halts

See Section 5.2, Reporting Problems.

5.4 Recommendations

Not applicable.

Appendix A Format Definitions

A.1 Preconfigured Parameter Aspect Types

The current version of the Parameter Builder contains the following preconfigured Parameter Aspect Types. Additional Parameter Aspect Types have to be configured.

A.1.1 Fieldbus Parameter Aspect Type

Table A-1. Fieldbus Parameter

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
FieldbusParameter	STATION	Byte	0	
FieldbusParameter	BUS	Byte	0	
FieldbusParameter	TYPE	Text	20	
FieldbusParameter	IMPL	Byte	0	
FieldbusParameter	BUSNO	Byte	0	
FieldbusParameter	STNNO	Byte	0	
FieldbusParameter	POS_I	Byte	0	
FieldbusParameter	SPOS_I	Byte	0	
FieldbusParameter	MASTER	Byte	0	

A.1.2 Station Parameter Aspect Type

Table A-2. Station Parameter

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
StationParameter	BUS	Byte	0	
StationParameter	STATION	Byte	0	
StationParameter	TYPE	Text	20	

A.1.3 Board/ Module Parameter Aspect Type

Table A-3. Board/Module Parameter

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
BoardParameter	STATION	Byte	0	
BoardParameter	BUS	Byte	0	
BoardParameter	POSITION	Byte	0	
BoardParameter	TYPE	Text	20	
BoardParameter	IMPL	Byte	0	
BoardParameter	CONV_PAR	Text	20	
BoardParameter	SCANT	Text	20	

A.1.4 Signal Parameter Aspect Type

Table A-4. Signal Parameter

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
SignalParameter	ACT	Byte	0	
SignalParameter	UNIT	Text	6	
SignalParameter	RANGEMAX	Double	0	
SignalParameter	RANGEMIN	Double	0	
SignalParameter	CONV_PAR	Text	20	
SignalParameter	HI_LIM2	Double	0	
SignalParameter	HI_LIM1	Double	0	
SignalParameter	LO_LIM1	Double	0	
SignalParameter	LO_LIM2	Double	0	
SignalParameter	PROC_SEC	Long Integer	0	
SignalParameter	NORM_POS	Byte	0	
SignalParameter	NORM_TR	Byte	0	

Table A-4. Signal Parameter (Continued)

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
SignalParameter	CLASS	Long Integer	0	
SignalParameter	VALUE_TR	Long Integer	0	
SignalParameter	DEC	Long Integer	0	
SignalParameter	RESTART	Text	5	
SignalParameter	STVAL	Byte	0	

A.1.5 Process Object Parameter Aspect Type

Table A-5. Process Object Parameter

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
ProcessObjectParameter	USED_IN_GROUP	Text	1	
ProcessObjectParameter	USED_IN_GROUP_EXT	Text	1	
ProcessObjectParameter	GROUP	Text	10	
ProcessObjectParameter	START1_ORD_NR	Text	10	
ProcessObjectParameter	STOP1_ORD_NR	Text	10	
ProcessObjectParameter	START2_ORD_NR	Text	10	
ProcessObjectParameter	STOP2_ORD_NR	Text	10	
ProcessObjectParameter	IA_SIGNAL	Text	20	
ProcessObjectParameter	IA_TXT	Text	20	
ProcessObjectParameter	IB1_SIGNAL	Text	20	
ProcessObjectParameter	IB1_TXT	Text	20	
ProcessObjectParameter	IB2_SIGNAL	Text	20	
ProcessObjectParameter	IB2_TXT	Text	20	
ProcessObjectParameter	IB3_SIGNAL	Text	20	
ProcessObjectParameter	IB3_TXT	Text	20	
ProcessObjectParameter	IB4_SIGNAL	Text	20	

Table A-5. Process Object Parameter (Continued)

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
ProcessObjectParameter	IB4_TXT	Text	20	
ProcessObjectParameter	IC2_SIGNAL	Text	20	
ProcessObjectParameter	IC2_TXT	Text	20	
ProcessObjectParameter	PROC_SECTION_NR	Text	2	
ProcessObjectParameter	REMOTE_IO	Text	10	
ProcessObjectParameter	IO_UNIT	Text	10	
ProcessObjectParameter	M1_USED	Text	1	
ProcessObjectParameter	M1_USED_EXT	Text	20	
ProcessObjectParameter	M2_USED	Text	1	
ProcessObjectParameter	M2_USED_EXT	Text	20	
ProcessObjectParameter	M3_USED	Text	1	
ProcessObjectParameter	M3_USED_EXT	Text	20	
ProcessObjectParameter	M4_USED	Text	1	
ProcessObjectParameter	M4_USED_EXT	Text	20	
ProcessObjectParameter	FB1_USED	Text	1	
ProcessObjectParameter	FB1_USED_EXT	Text	20	
ProcessObjectParameter	FB2_USED	Text	1	
ProcessObjectParameter	FB2_USED_EXT	Text	20	
ProcessObjectParameter	MOTOR_CURRENT	Text	1	
ProcessObjectParameter	MOTOR_CURRENT_EXT	Text	20	
ProcessObjectParameter	NOMINAL_LOAD	Text	10	
ProcessObjectParameter	BACK_SIGNAL	Text	1	
ProcessObjectParameter	BACK_SIGNAL_EXT	Text	20	
ProcessObjectParameter	COMMON_BACK_SIGNAL	Text	1	
ProcessObjectParameter	COMMON_BACK_SIGNAL_EXT	Text	20	
ProcessObjectParameter	USE_FLT1	Text	1	
ProcessObjectParameter	USE_FLT1_EXT	Text	20	
ProcessObjectParameter	USE_LCB	Text	1	
ProcessObjectParameter	USE_LCB_EXT	Text	20	

Table A-5. Process Object Parameter (Continued)

AspectTypeName	AspectTypeAttribute	DataType	Length	Comment
ProcessObjectParameter	CLOSE_OUTPUT	Text	1	
ProcessObjectParameter	CLOSE_OUTPUT_EXT	Text	20	
ProcessObjectParameter	LSW_OPN	Text	20	
ProcessObjectParameter	LSW_CLS	Text	20	
ProcessObjectParameter	PU_CTRL	Text	1	
ProcessObjectParameter	TRQ_OPN_USED	Text	1	
ProcessObjectParameter	TRQ_OPN_USED_EXT	Text	1	
ProcessObjectParameter	TRQ_CLS_USED	Text	1	
ProcessObjectParameter	TRQ_CLS_USED_EXT	Text	1	
ProcessObjectParameter	ACT_POS	Text	20	

A.2 How to Define References

References are used to access single parameter data or Structure Builder data from outside. This is typically done in

- Type circuits of the AMPL Control Configuration
- Documents managed by the Document Builder.

The reference is a string that contains all information, necessary to identify and access one single attribute of a Parameter aspect type or dedicated Structure Builder aspect type in a project. Those can be accessed by a reference.

Three different formats of references are supported:

1. Relative reference to the same object
2. Relative reference to another (related) object
   - by name (CHILD and PARENT)
   - by absolute reference designation (CHILD only)
3. Absolute reference to any object.

If a referenced is invalid and cannot be resolved an error message is displayed and no value is returned.
If the reference is not unique no error is displayed. The value of the first hit is returned.

A.2.1 Relative Reference to the Same Object

A relative reference to the same object has the format:

$$REF.*aspect description.attribute name*

$$REF
Is a constant token where each reference starts with.

aspect description
The reference points to an aspect of the same object. This aspect is identified by its aspect description. Classes of aspect descriptions are:

- Structure Builder (STB) aspect types like
  - Basic Property
    description default values: Customer ID, Instance ID and so on
  - Structure
    description default values: = Function Structure, == OCS-Structure and so on
- Parameter aspect types like
  - BoardParameter
    description default value: BoardParameter
  - SignalParameter
    description default value: SignalParameter
  - and the rest of Parameter aspect types created in actual project.

attribute name
The aspect (identified before by *aspect description*) has an attribute, which is identified by its attribute name. Its value should be fetched.

For aspect type Basic Property the following attribute names are supported:

- Asp_Descr
  description of aspect
- Name
  name of Basic Property
- NameCategory
  category name of Basic Property (e.g. Customer ID, Instance ID, ..)
- ObjectType
  type of object (e.g. Area, Bus, Signal,...)

For aspect type Structure the following attribute names are supported:

- ABS or ARD
  absolute reference designation
- RRD
  relative reference designation
- Prefix
  prefix of reference designation (e.g. =, ==, ++ and so on)

- RDCTRL
  specifies inheritance of reference designation (e.g. sign < stops inheritance)
- StrCategory
  structure category (e.g. = Function Structure, ++ Location Structure,...).

For Parameter aspect types is every attribute name of each Parameter aspect type prepared in current project usable.

In the example below the attribute name is *IMPL*. It is one attribute of parameter aspect type BoardParameter.

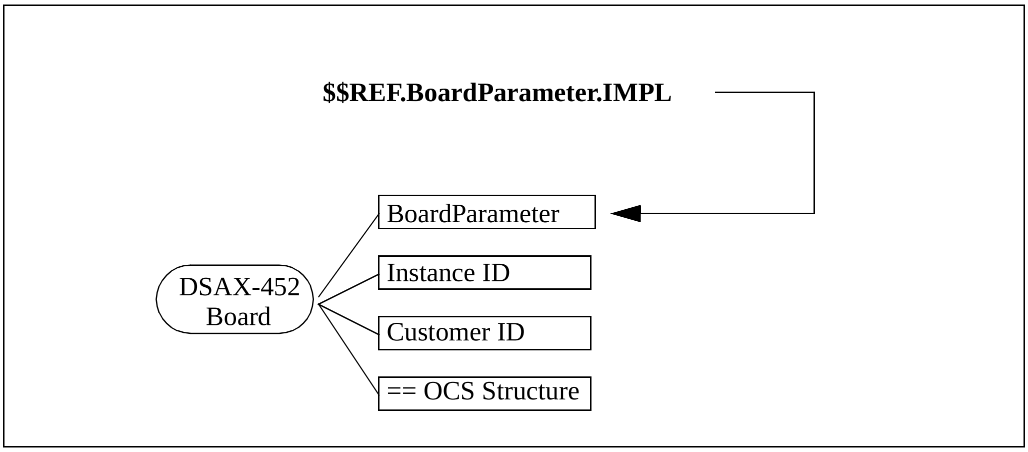


Figure A-1. Example - Relative Reference to the Same Object

A.2.2 Relative Reference to Another (Related) Object

A relative reference to another (related) object has the format either:
identified by name:

$$REF.*structure.RelPos.Categrory.Oname.aspect description.attribute name*

or by absolute reference designation*:*

$$REF.*structure.RelPos.ARD.AbsRefdes.aspect description.attribute name*

$$REF
Is a constant token where each reference starts with.

structure
Name of the structure, where the two objects are in parent/child relationship.
In the example below it is the *= Function Structure*.

RelPos
Describes the relationship within structure. Valid values are either *CHILD*, which means that the target object is placed somewhere below the starting object, or *PARENT*, the reverse case somewhere above.
In the example below the starting object is *motgrp1*.

Identified by name

- ***Category***
  Is the basic property category of the target object name.
  In the example below it is *Customer ID*.
- ***Oname***
  Name of the target object the reference points to..
  In the example below it is *mot1*.

Identified by absolute reference designation

- ***ARD***
  keyword indicates that the next value will be read as absolute reference designation (dot rule assumed).
- ***AbsRefDes***
  Designation must be enclosed with quotation marks. The wildcards character "*" in front of designation is possible. * stands for one or more then one relative reference designations.

aspect description
Identifies one aspect of the target object by the aspect description. The set of valid aspect descriptions and attribute names are already described in Appendix A.2.1, Format Definitions
In the example below the reference points to the aspect, which has the value *ProcessObjectParameter* in its aspect description.

attribute name
The aspect has an attribute, which is identified by its attribute name. Its value should be fetched. In the example below the attribute name is *GROUP*. It is one attribute of parameter type ProcessObjectParameter. For example the motgrp1 Advant object contain a document aspect which includes the reference string
$$REF.= Function Structure.CHILD.Cutomer ID.mot1.ProcessObjectParameter.GROUP
or identified by absolute reference designation
$$REF.= Function Structure.CHILD.ARD."*.T1".ProcessObjectParameter.GROUP
to present the GROUP value.

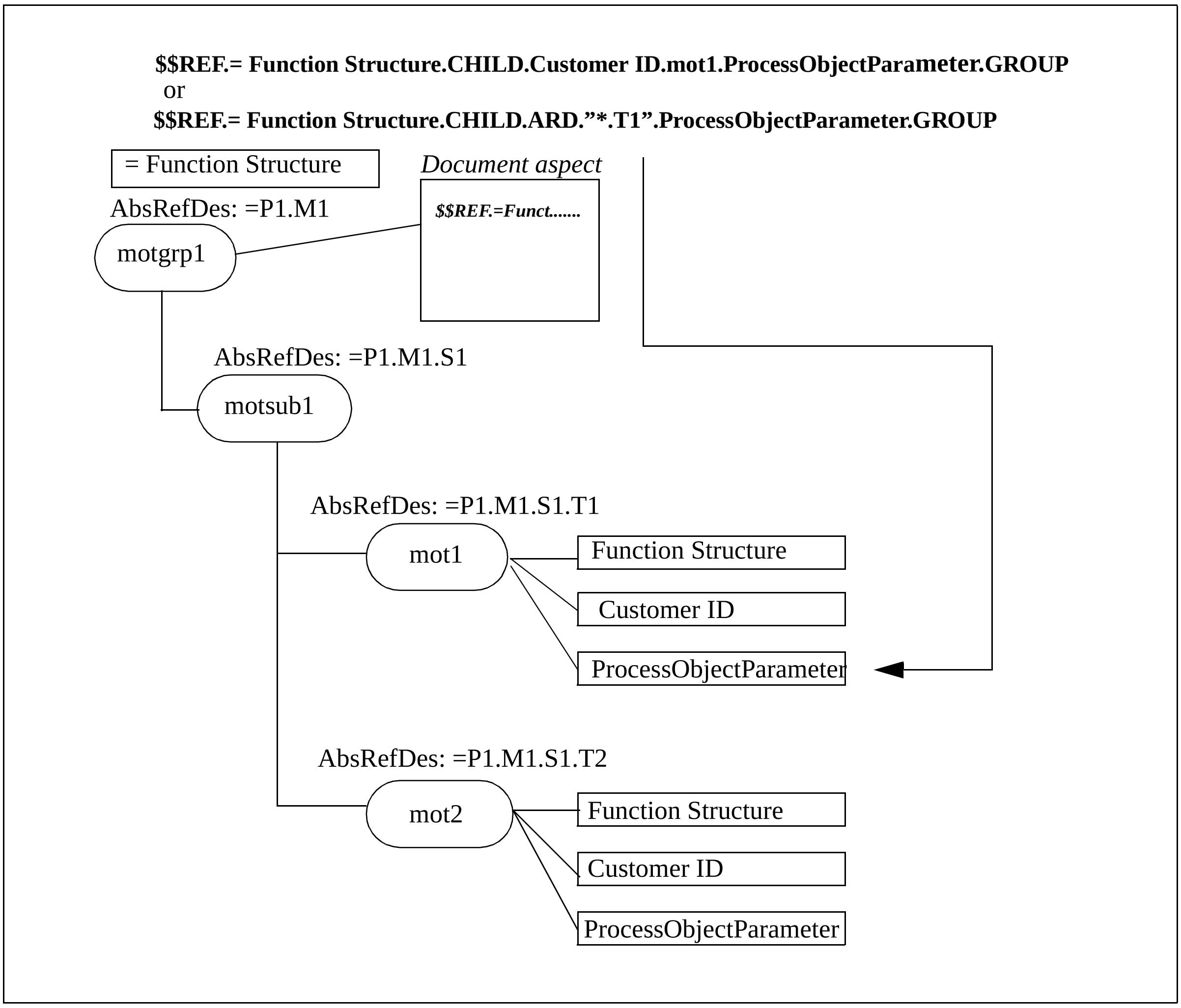


Figure A-2. Example - Parameter Reference to Another (Related) Child Object

A.2.3 Absolute Reference to Any Object

An absolute reference to any object has the format:

$$REF.*Categrory.Oname.aspect description.attribute name*

$$REF
Is a constant token where each reference starts with.

Category
Is the basic property category of the target objects name.
In the example below it is *Customer ID.*

Oname
Name of the target object the reference points to.
In the example below it is *DSAX-452.*

aspect description
Identifies one aspect of the target object by the aspect description. The set of valid aspect descriptions and attribute names are already described in Appendix A.2.1, Format Definitions. In the example below the reference points to the aspect, which has the value *BoardParameter* in its aspect description.

attribute name
The aspect (identified before by *aspect description*) has an attribute, which is identified by its attribute name. Its value should be fetched.
In the example below the attribute name is *SCANT.* It is one attribute of parameter type Board-Parameter.

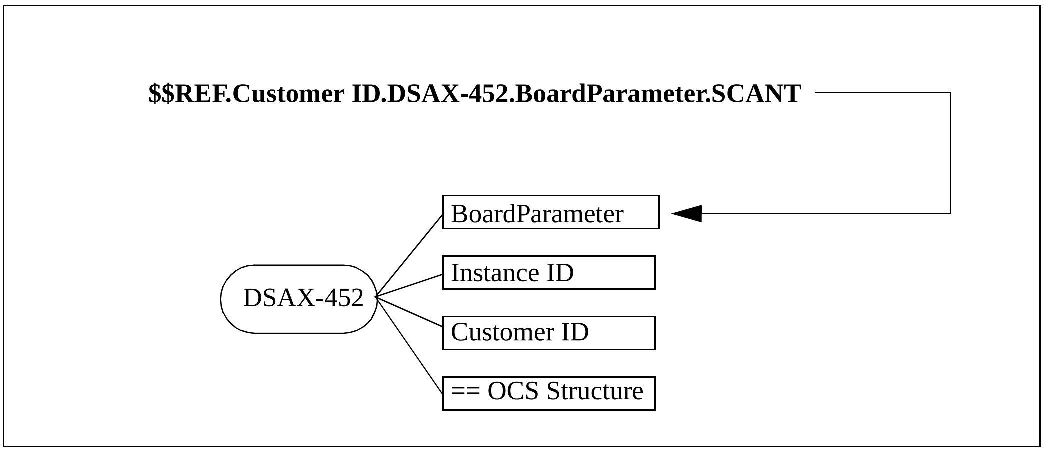


Figure A-3. Example - Absolute Parameter Reference to Any Object.

Appendix B Glossary of Terms

Administration

All activities covering user registration and creation, renaming, backup and restore, and deletion of projects are summarized under the topic administration or project administration.

AdvaBuild_engineers

Name of a Windows NT group created for working with **AdvaBuild Engineering Software**. Via this group the access rights of users are controlled. All members of this group can access project data. Therefore it is necessary to be assigned to this group.

Creation and deletion of projects is also controlled via the **AdvaBuild_engineers** group. But in addition you need the administrator password for the database user **EOSTB**.

Aspect

There are often several ways to look at an Advant object, depending on how you want to handle it. When for example an Advant object is representing a valve, then you might be interested in a drawing of that valve or in its loop. Then you may create with a corresponding Builder an CAEE aspect for the drawing and a control function aspect for the loop. The aspect stores the corresponding data. An aspect is an instance of an aspect type. Normally a Builder supplies some aspect types it is able to work on. You can start a Builder on an aspect by opening one of its views.

Aspect Attribute

An attribute holds descriptive data of an aspect, for example the attribute BUS of the Parameter Aspect Type BoardParameter.

Aspect System

The Aspect Request Broker communicates with a Builder via a so-called Aspect System Interface. When some Builders share an aspect system interface, these Builders are referred to as an aspect system. Therefore, an aspect system may consist of one or several Builders.

Aspect Type

Each Builder supplies several aspect types. Examples of aspect types are basic property aspect, structure aspect, or parameter aspect and its sub types or category like BoardParameter, SignalParameter, StationParameter and so an. An aspect type is designed for special tasks. These tasks will be performed by the corresponding Builders or aspect systems. An aspect type works as a template for the creation of aspects.

Attributes

See Aspect Attributes.

Basic Property Aspect Type

The basic property aspect type is one of the aspect types supplied by **Structure Builder**. It holds a name and the type of an Advant object. As an Advant object may have several names, it may have several basic property aspects. You can refer to a specific one with the help of its name category, for example with Customer ID.

An Advant object may have for example a customer ID “Turbine A” and a location ID “Turbine A 2nd floor”.

Builder

A Builder is an application that can perform dedicated tasks on aspects. Builders can cooperate which each other via the **Aspect Request Broker**. Examples of Builders are: Structure Builder, Function Chart Builder, Parameter Builder.

Category

A category is a subtype of an aspect type. The name category for example is a subtype of a basic property aspect.

Child Object

Advant objects can be visualized in a pool or in a structure window. In a structure Advant objects may have a parent-child relation. A parent object may contain several child objects. In this way you can create a hierarchy of Advant objects. Child objects can be hidden when you compress the parent object.

Cooperating Mode

Builders can work in cooperating or independent mode. In cooperating mode Builders can communicate via the **Aspect Request Broker**. They are started by the **Structure Builder**, when you open a view of a selected aspect.

Copy

You can copy Advant objects between projects, structures, or within structures. In each case copy creates an additional Advant object.

Compressed Structure

For parent objects you can specify that the structure shall be displayed in compressed form. Then all successive child elements are hidden.

Database

The data of **Parameter Builder** are stored in an **Access** database. The data are collected by a project view. Add project create the Parameter Builder database will be created; each project has it own database. The project creation can be performed with the **Project | Administration** menu command.

Data Sheet

Besides the data stored in the aspect registry, Structure Builder stores additional data in an own format. These data can be presented in data sheets. For data sheets the same spread sheet representation as for the aspect registry is used.

Drag and Drop

You can move, copy, or place Advant objects between or within structures by grabbing an object with the mouse and positioning it at the new location while the left mouse button is pressed. When you release the mouse button, the object will be copied, placed or moved, depending on additional keys you might have pressed during the operation: Alt: move; Ctrl: Copy; Shift: place. If no additional key is pressed, a default operation will be performed, which depends on source and destination.

Advant Object

Advant object Technology offers a methodology to model real world objects with Advant objects, aspects, and views. All entities you work with during the engineering process, like main plant equipment, components and devices, information entities like signals, or location entities like areas are referred to as Advant objects. Each Advant object may have several aspects.

EOSTB

EOSTB is a special database user which has the access rights to create and delete projects and to change user passwords. For project administration you have to log into the system as EOSTB user.

Expanded Structure

For parent objects you can specify that the structure shall be displayed in expanded form. Then successive child elements on the next lower level or on all levels are visible.

Menu

Menus are part of the menu bar. When you click with the mouse on a menu bar entry, the menu will be displayed, which shows one or several menu commands or a submenu. If you click on a menu command, the command will be executed.

Menu Bar

The menu bar is a bar below the main window's title bar. It holds the menu entries.

Move

To change the location of an Advant object in one of your structures, you can move it to another location in the same or in another structure.

Object

See Advant object.

Object Name

The name of an Advant object.

Object Type

The type of an Advant object. When you create a new Advant object, you can select among several types.

Object Window

You will see the object window when you issue the **Object | Open** menu command or when you double-click an Advant object. In the object window you see the aspects and views of the selected Advant object.

Parent Object

Advant objects can be visualized in a pool or in a structure window. In a structure Advant objects may have a parent-child relation. A parent object may contain several child objects. In this way you can create a hierarchy of Advant objects.

Paste

You can insert Advant objects, that you have previously copied to the clipboard with the **Edit | Copy** menu command, at the currently selected object.

Place

You can place an Advant object somewhere in a structure. When you place an Advant object, you do not create a new object. The object gets a new structure aspect containing the information about the new structure position.

Pool

The pool is a list of all Advant objects. The order the elements are listed in that list is arbitrary.

Prefix

The prefix determines the type of a reference designation. Some prefixes are standardized:

=	Function reference designation
-	Product reference designation
+	Mounting location reference designation
++	Topological location reference designation

Within AdvaBuild Engineering Software the following designation is used:

==	OCS item designation

Besides these predefined prefixes, you are free to use your own prefixes.

Further on, the prefix will be used to concatenate relative reference designations.

Project Administration

Project administration covers all activities to administer projects, to assign database users, and to assign users to projects.

Project Global Data Aspect Type

At project creation time a project global data aspect will be created as an aspect of the project object. You cannot delete this aspect nor create another one. The project global data aspect type stores general information about a project.

Reference Designation

A reference designation designates an Advant object in a structure unambiguously (not checked by the system) with respect to its parent objects. Absolute reference designations consists of a concatenation of all relative reference designations.

Selected Pool

A selected pool is a special pool that is the result of a filter. It is a subset of the pool.

SQL Statement

SQL statements are used to specify filters for Advant objects. They are stored in files which you can select.

Status Bar

The status bar is a bar at the bottom of the application's main window. It holds some status information like keyboard settings.

Structure

Advant objects can be related to each other hierarchically. These relations are visualized in a structure window. You can create several structures depending on your needs. Some predefined structures are available.

Structure Aspect Type

When you locate an Advant object in a structure, it will get an corresponding structure aspect, which will hold the information about the position within the structure and the object's parent object.

Substructure

The part of a structure that holds all child objects of a certain (parent) object. It is also referred to as subtree.

Toolbar

The toolbar is a bar locate directly below the menu bar. It offers a quick access to some of the most commonly used menu commands.

User

A member of a project must be known by the database as user. Members of the Windows NT group **AdvaBuild_engineers** can be assigned as database users and can be mapped to projects.

View

An aspect may have several views. You can start a Builder on an aspect by opening one of its views. A node control software aspect as among others for example a node function chart and a node structure function chart view.

INDEX

A
Absolute Reference to any Object A-9
Add new Aspect Type 3-21
Advant object 3-6, 3-18
Apply Filter 3-8

B
Board/ Module Parameter Aspect Type A-2
Building rule- object name 3-16

C
Configuration of Parameter Aspect Types 3-19
Configuration of Reports 3-45
Conventions 1-2
Copy a Parameter Aspect 3-4
Create a new Parameter Aspect 3-3
Creation of Project 3-17

D
Data Type Converting Rules 3-36
Database Error Messages 5-1
Datasheet 3-7, 3-53
Delete a Parameter Aspect 3-4
Delete Aspect Type 3-34
Design View 3-45
Directory Structures 3-1

E
Export Aspect Type Definition 3-40

F
Fieldbus Parameter Aspect Type A-1
Filter 3-9
Filter dialog 3-54
Find Criteria 3-12
Forms Designer Dialog 3-56
Formsheet 3-7, 3-51
Full Check 3-43

H
How to Define Parameter References A-5
How to Upgrade an Old Project 3-2

I
Import Aspect Type Definition 3-41
Import Parameter Aspects 3-5

L
Layout of the Aspect Type Definition File 3-41
Local Parameter Builder Configuration 3-42

M
Menus 3-51
Modify Attributes 3-12
Modify existing Aspect Type 3-35

O
Open a Parameter Aspect 3-5

P
PAR Duplicate Behaviour 3-43
Parameter Reference 1-4
Print a Parameter Aspect 3-5
Print Aspect Type Definition 3-42
Print Preview 3-45
Print Preview Dialog 3-55
Process Object Parameter Aspect Type A-3
Product Guide 1-1
Product Structure 1-3

R
Records 3-55
Related Documentation 1-3
Relative Reference to another Object A-7
Relative Reference to the same Object A-6
Replace Function 3-13
Report Designer Dialog 3-56

S
Search for Aspect Type in other Projects 3-37
Search for Parameter Values 3-9
Signal Parameter Aspect Type A-2
Sort Ascending 3-11
Sort Descending 3-11
Station Parameter Aspect Type A-1

T
Toolbars 3-56
Trace file 3-44

3BSE013930R201

October 1999